

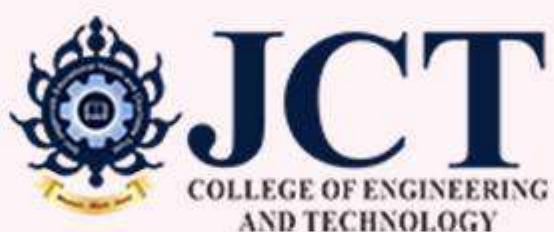


Proceedings

of the

7th International Conference on Inventive Systems and Control (ICISC 2023)

30-31, January 2023 | Coimbatore, India



organized by

JCT College of Engineering and Technology
Coimbatore, India

Table of Contents

#	Manuscript Title / Authors
1	Optimal placement of phasor measurement units considering channel limits under various contingencies <i>Banumalar K, Manikandan B V, Muralidharan S</i>
2	Adaptive Deep Recurrent Neural Network Based COVID-19 Healthcare Data Prediction for Early Risk Prediction <i>Asha A, Dhiyanesh B, Kiruthiga G, Shakkeera L, Sharmasth Vali Y, Karthick K</i>
3	Maximum Decision Support Regression Based Advance Secure Data Encrypt Transmission For Healthcare Data Sharing In The Cloud Computing <i>Anusuya V, Bejoy B J, Ramkumar M, Shanmugaraja P, Dhiyanesh B, Kiruthiga G</i>
4	Routing Integrity Mechanism To Prevent Wormhole Attacks In Vehicular Adhoc Networks <i>Prathap Kumar R, U Srilakshmi, Ganesh Reddy K</i>
5	A detailed analysis on spam emails and detection using Machine Learning algorithms <i>Razia Sulthana A, Avani Verma, Jaithunbi A K</i>
6	Advanced Encryption Standard based Encryption for Secured Transmission of Data in Cognitive Radio with Multi-Channels <i>Kiran P More, Rajendrakumar A Patil</i>
7	MULTI ENERGY-HARVESTING SMART WATER METER DESIGN FOR UNDERGROUND WATER PIPELINE LEAKAGE DETECTION <i>Hari Prakash Athinarayanan, Muthupavithran Selvam</i>
8	Tensor Flow Model with Hybrid Optimization Algorithm for solving Vehicle Routing Problem <i>Jai Keerthy Chowlur Revanna, Nushwan Yousif B Al-Nakash</i>
9	Authentication key generator for data sharing on cloud- A Review <i>Santhosh Krishna B V, B Rajalakshmi, Esikala Nithish Manikrishna, Gundre SaiSruthi, Gangireddy Ramyasri, Ashok K</i>
10	Restaurant Quality Analysis: A Machine Learning Approach <i>Rohit B Diwane, Dr. Kavita S Oza, Dr. Varsha P Desai</i>



11	Towards the Implementation of Traffic Engineering in SDN: A Practical Approach <i>Prabu U, Geetha V</i>
12	Real time intrusion detection in connected autonomous vehicles <i>Anjanee Kumar, Tanmoy Kanti Das</i>
13	Physical Architecture of Linear Feedback Shift Register using Clock Tree Synthesis for Cyber-Physical System <i>B Muthu Nisha, V Nithya, J Selvakumar</i>
14	Detection of Arrhythmia via Electrical Activity of the Heart using AI techniques <i>Pramitha J, Anitha Mary X</i>
15	Utilizing Deep Convolutional Neural Networks for Image-Based Plant Disease Detection <i>Saha Reno, Marzia Khan Turna, Sheikh Tasfia, Md. Abir, Anusha Aziz</i>
16	Detection of Covid-19 using an Infrared Fever Screening System (IFSS) based on Deep Learning Technology <i>V Muthu, S Kavitha</i>
17	Improving the power quality of wind turbines under unbalance voltage conditions using the SMC approach. <i>HARIPRASAD B, Dr. Sreenivasan G, Mahesh Kumar D, Lakshmi Swapna T</i>
18	NAML- A Novel Approach of Machine Learning Implementation in the Hospitality Industry <i>Ashwin C S, Dr. Sheela Thavasi, Rangarajan KR</i>
19	CISUM: Novel research on Cloud Computing Simulators and Future Scope for Computational Research <i>Ashwin C S, VKG Kalaiselvi, Rangarajan KR</i>
20	A Hybrid Model Built on VGG16 and Random Forest Algorithm for Land Classification <i>Mohammed Junaid Ahmed, Ashutosh Satapathy, Ch. Raga Madhuri, K Yashwanth Chowdary, A Naveen Sai</i>
21	Facial Expression Recognition using Transfer Learning with RESNET50 <i>Shantala S Hiremath, Jayaprada Hiremath, Vaishanvi V Kulkarni, Harshit B C, Sujith Kumar, Mrutyunjaya S Hiremath</i>



22	Classification of Microorganisms from Sparsely Limited Data using a Proposed Deep Learning Ensemble <i>Gautam Chettiar, Amogh Shukla, Hemprasad Patil, Sumit Jindal</i>
23	Assiduous Study of the Hyper-parameters' Influence on CNN using COVID-19 CT Images <i>Srinivasa L Chakravarthy, Varun Mallela, Vedula Sai Sarvanth, Rohith Sunkara, Srimurari Dacheppalli</i>
24	Academic Dishonesty Detection in Exams using Pose Extraction <i>Dhanush Binu, Dr. Sivaiah Bellamkonda</i>
25	Dynamic Wireless Charging System for Electric Vehicles based on Segmented Zinc Alloy Plates <i>Anurag A Gadgil, Arya Bairoliya, Febin Daya J L, Balamurugan P</i>
26	Iterative Refinement vs Generative Adversarial Networks for Super-Resolution toward Licence Plate Detection. <i>Alden Boby, Dane Brown, James Connan</i>
27	An IoT-Based Smart Health Monitoring System <i>R Lakshmi, M Mridula, G Sri Gayathri, V Srividhyasakthi</i>
28	Multi Stage Fruit Grading System <i>Joel J Sebastian, J Krishnanunni, Joveal K Johnson, Ashik Mujeeb, Anjali S, Vinny Pious</i>
29	Sentiment Analysis on feedback data of E-commerce products based on NLP <i>Dr. K Sumathi, Dr. Kundhavai Santharam</i>
30	Univariate Individual Household Energy Forecasting by Tuned Long Short-Term Memory Network <i>Marko Stankovic, Luka Jovanovic, Milos Antonijevic, Aleksandra Bozovic, Nebojsa Bacanin, Miodrag Zivkovic</i>
31	Interpreting Doctor's Handwritten Prescription Using Deep Learning Techniques <i>Rizwanullah Mohammad, Ajay Kumar Varma Nagaraju, Suneetha Manne</i>
32	Abductive Inference of Conclusions with Check of Additional Premises Literals Correctness Interpretation <i>Vasily Meltsov, Dmitry Strabykin, Alexander Krutikov</i>
33	An Investigation and Observational Remarks on Conventional Sign Language Recognition <i>Thouseef Ulla Khan, Dileep M R</i>



34	Exploratory Project of Digital Twin Technology in Cyber-physical Systems <i>Irony Nunes de Oliveira, Hanameel Carlos Vieira Gomes, Kelipys Da Silva Firmino, Antonio Eduardo Carrilho Da Cunha, Cicero Roberto Garcez</i>
35	ABSTRACTIVE TEXT SUMMARIZATION FOR TAMIL LANGUAGE USING m-T5 <i>C Saraswathi, V Prinitha, J Briskilal</i>
36	Sentiment Component Extraction from Dependency Parse for Hindi <i>Remya Panicker, Ramchandra Bhavsar, B V Pawar</i>
37	Change Detection Algorithm for Vegetation Mapping using Multispectral Image Processing <i>Neelam B V D Soujitha, Mohammad Neelofar Jaha, Mahali Tirumala Raju, Kakamanu Christy Victor, Radhesyam Vaddi</i>
38	Real Time Sign Language Detection Using OpenCV <i>Pavan Kumar Meka, Yesu Raju Parusu, Radhesyam Vaddi</i>
39	Social Media as the Most Effective Means of Business Promotion Today With Using Social Media Advertising Technology <i>ALDO SAPUTRA, INGWEN TANNARIS, KELBY HUBERT, Ford Lumban Gaol, Tokuro Matsuo</i>
40	Investing in Products with the Greatest Demand on Online Stores during the Pandemic <i>Titan Hassya, Muhammad Fauzi Hanif, Alvian, Ford Lumban Gaol, Tokuro Matsuo</i>
41	The Adoption of Artificial Intelligence Technology in Parking Security System <i>Muhammad Rifqi Alhafizh, Albertus Baskara Yunandito Adriawan, Darryl Fernaldy, Ford Lumban Gaol, Tokuro Matsuo</i>
42	A Self-Organizing Map with Neural Networks Classifier Technique for Face and Handwritten Signature Recognition System in DIP <i>Karthi V, Nithyasai S, Parthasarathy P, Arun Kumar U</i>
43	A NOVEL DYNAMIC ENERGY AMENDMENT CONTROL FOR SOLAR FED BRUSHLESS DC MOTOR FOR WATER PUMPING SYSTEM <i>Arun Kumar U, Chandraguptamauryan K S, Chandru K, Rajan Babu W</i>
44	Arabic Sentiment Analysis of YouTube Comments using Deep Learning Model <i>Mohammed Alkoli, B Sharada</i>



45	Learning Movement Patterns for Improving the Skills of Beginner Level Players in Competitive MOBAs <i>Dane Brown, Jonah Bischof</i>
46	Darknet Traffic Detection Using Histogram-Based Gradient Boosting <i>Dane Brown, Chikondi Sepula</i>
47	Efficient Plant Disease Detection and Classification for Android <i>Dane Brown, Sifisokuhle Mazibuko</i>
48	Text-to-Speech and Speech-to-Text Converter - Voice Assistant <i>Sagar Janokar, Soham Ratnaparkhi, Manas Rathi, Alkesh Rathod</i>
49	CNN combined with FC Classifier to combat Artificial Penta-digit text-based Captcha <i>S Abhishek, Sanjana S, Mahima Chowdary Mannava, Anjali T</i>
50	Basil Leaf Disease Detectionand Classification using Customized Convolutional Neural Network <i>Deepak Mane, Sunil Sangve, Shaila Jadhav, Disha Patil, Rohan Kakde, Varad Marudwar</i>
51	Segmentation of Brain Tumours from MRI Images Using CNN <i>Ilanga, D; Sulthana, R</i>
52	Distributed Training of Large Scale Deep Learning Models in Commodity Hardware <i>Jubaer Ahmad, Tahsin Elahi Navin, Fahim Al Awsaf, Md. Yasir Arafat, Md. Shahadat Hossain, Md. Motaharul Islam</i>
53	Blockchain-Driven Cloud Service: A Survey <i>Hamed Taherdoost</i>
54	Interaction analysis of a multivariable process in a manufacturing industry - a review <i>P K Juneja, T Anand, P Saini, Rajeev Gupta, F S Gill, S Sunori</i>
55	Machine Learning Approaches for Educational Data Mining <i>Mahesh Bapusaheb Toradmal, Dr. Mita Mehta, Dr. Smita Mehendale</i>
56	Traffic Surveillance and Vehicle Detection YOLO and MobileNet-based ML pipeline transfer learning <i>Prof. Rakhi Bharadwaj, Aditya Thombre, Umesh Patekar, Yash Gaikwad, Sushil Suri</i>



57	Perceptors: A Real Time Object Detection System with Voice Feedback and Distance Approximation for Blind <i>Prof. Rakhi Bhardwaj, Harshal Sonawane, Manasi Patil, Shashank Patil, Vedant Jadhav</i>
58	Automated Histogram Binning Based Fuzzy K-Means Clustering for COVID-19 Chest CT Image Segmentation <i>S Nivetha, H Hannah Inbarani</i>
59	An Effective Multi-Exposure Fusion Approach Using Exposure Correction and Recursive Filter <i>Jishnu C R, Vishnukumar S</i>
60	Resume Analysis Using NLP <i>Rakhi Bharadwaj, Divya Mahajan, Meenal Bharsakle, Kashish Meshram, Himaja Pujari</i>
61	Estimation of Queuing System Incoming Flow Intensity <i>Alexander Zyulkov, Yury Kutoyants, Svyatoslav Perelevsky, Larisa Korableva</i>
62	Statistical Characteristics of the Adjacent Information Signals Amplitude Ratio <i>Oleg Chernoyarov, Alexey Glushkov, Vladimir Litvinenko, Alexandra Salnikova, Kaung Myat San</i>
63	Predicting Severity Levels of Parkinson's Disease from Telemonitoring Voice Data <i>Aryan Vats, Aryan Blouria, Sasikala R</i>
64	Attendance Management System Using Face Recognition <i>I G Vishnu Vardhan, M Ajay Kumar, P Nithin Kalyan, V Spandana, Dr. J Ebenezer</i>
65	Diabetes Classification Using ML Algorithms <i>G G Rajput, Ashvini Alashetty</i>

Optimal placement of phasor measurement units considering channel limits under various contingencies

*Banumalar.K¹, Manikandan.B.V¹ and Muralidharan. S¹

¹Department of Electrical and Electronics Engineering,
Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India

* Corresponding Author: kbanumalar@mepcoeng.ac.in

Abstract. Phasor measurement units (PMUs) are becoming increasingly important for real-time monitoring, protection, analysis, and control of modern power systems. The sine cosine algorithm (SCA) is proposed in this work to solve the optimal PMUs placement problem formulation under various restrictions, including the presence of a zero injection bus, channel limits, single line outage, and single PMU loss. The objective is to reduce the number of PMU installations while increasing measurement redundancy, under full network observability. The proposed OPP problem was tested and compared against existing techniques on IEEE 14-bus, IEEE 30-bus, IEEE 39-bus, IEEE 57-bus, and IEEE 118-bus test systems. When compared to existing literature, the results show that the suggested algorithm is simple and accurate in determining the number of PMU installations that are less or equal under various contingencies.

Keywords: Phasor measurement units, Channel limits, Measurement redundancy, Zero injection buses, Single line outage, Single PMU loss, Zero injection buses.

1 Introduction

Phasor measurement units (PMUs) are frequently utilised in electrical system monitoring, safety, and control. Cause appropriate PMU setup in the network to improve full observability. However, due to financial constraints, the number of PMUs must be kept to a minimum [1]. Zero injection buses are initially considered to reduce PMU requirements while maintaining network observability [2-4]. Proper PMU issue formulation modelling eliminates contingencies like single line interruption and single PMU loss, improving network dependability [5]. Aside from that, a few studies [6-7] explore full network observability when there is ZIB and PMU channel constraints under various scenarios.

Various methods have been reported to solve optimal placement and number of PMU in order to improve the system completely observable, including mathematical methods like integer linear programming (ILP) [8], binary ILP (BILP) [9], as well as meta heuristic methods like tabu search [10], genetic algorithm[11], firefly algorithm [12], particle swarm optimization [13], mixed integer linear programming (MILP)[13]. The authour in [18-19] proposes an evaluation of the bit

error rate (BER) performance for a massive multi-input multi-output (M-MIMO) system using a spatial time shift keying (STSK) scheme over a three-dimensional (3-D) fading model.

In a prior work, we used the sine cosine algorithm (SCA) in a partial shading MPPT issue [14]. Following successful completion, an effort is made to solve optimal PMU placement using the sine cosine algorithm for four cases: base case, ZIBs, single PMU failure, and single line outage under various channel restrictions. The rest of this article is structured as follows: The mathematical formulation of the PMU placement problem with ZIBs and observability criteria, loss of one PMU, and single line outage are discussed in Section 2. Section 3 shows how to use the SCA technique to solve an optimal placement problem. The use of SCA to tackle the OPP problem is discussed in Section 4. Finally, in section 5, the simulation findings are reported, and section 6 concludes the results.

2 Problem Formulation

The goal of the PMU placement problem is to reduce the number of PMUs in an electric network, as stated in [15].

$$\text{Min } F(z) = \sum_{i=1}^N c_i z_i \quad (1)$$

Subjected to

$$f = A_{cm}Z \geq b \quad (2)$$

The cost (c_i), binary state (z_i) of the PMU deployed at bus I, connectivity matrix(A_{CM}) and observability of corresponding bus (f) can be defined as follows:

$$z_i = \begin{cases} 1, & \text{if PMU installed at bus } i \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$A_{CM} = [a_{ij}]_{N \times N} \quad (4)$$

$$Z = [x_1, \dots, x_N]_{N \times 1}^T \quad (5)$$

$$b = [1, \dots, 1]_{N \times 1}^T \quad (6)$$

$$f = [f_1, f_2, \dots, f_{N_B}]_{N \times 1}^T \quad (7)$$

2.1 Measurement redundancy

All of the buses in an electrical system must be observable for the system to be fully observable [15]. The following is a description of measurement redundancy (MR), which is an important aspect in monitoring the electrical network and solving the OPP problem.

$$\text{Measurement redundancy} = \sum (A_{cm}Z) \quad (8)$$

2.2 Zero injection bus

When the power system is observable and meets the following requirements [15], the zero injection bus minimises the number of PMU required.

- i. Unobserved bus are observed by observable ZIB which is linked to it by using KCL at ZIB.
- ii. If unobserved ZIBs are linked to all the observable buses, then ZIB is observable by using KCL.

2.3 Single PMU loss or single line outage

Any contingencies must not disrupt the electrical system's monitoring. Any single PMU failure caused the entire electrical network to go down. To solve this difficulty, all buses were observed twice to ensure that the system was completely observable [15], as follows:

$$b_{n \times 1} = [2 \ 2 \ 2 \ \dots \ \dots \ \dots \ J^T] \quad (9)$$

2.4 Channel limits

Manufacturers usually keep the channel number of PMUs fixed. A limited number of PMUs are employed to track the number of branch currents and bus voltages at any given moment. PMU numbers(N_k) change when channel numbers (L)change, and the two are inversely proportional. The channel limitations(C_k) are taken into account for solving this OPP problem, as shown in Equation [10].

$$C_k = \begin{cases} \frac{N_k!}{(N_k - L)!L!} \dots N_k > L \\ 1 \dots N_k \leq L \end{cases} \quad (10)$$

3 Optimal Placement of Phasor Measurement Unit

Sine cosine algorithm (SCA) is a stochastic optimization approach that proposes many initial random candidate solutions based on a mathematical model of sine and cosine functions [16]. The exploration and exploitation phases are the two phases of SCA. Exploration deals with random change in random solutions, while

exploitation deals with random selection to produce a set of solutions. The steps of SCA's operation are listed below. Using the equation to update the position during the exploration and exploitation phase,

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times |r_3 P_i^t - X_i^t|, & r_4 < 0.5 \\ X_i^t + r_1 \times \cos(r_2) \times |r_3 P_i^t - X_i^t|, & r_4 \geq 0.5 \end{cases} \quad (11)$$

$$r_1 = a - t \frac{a}{T} \quad (12)$$

where,

X_i^t -current position(i) at t^{th} iteration

P_i^t - position (i) of destination at t^{th} iteration,

r_4 - random number [0,1],

r_1 r_2 r_3 - random numbers,

t - current iteration,

T - maximum number of iterations, and

a - constant.

The parameter r_1 determines whether the next position is inside or outside of space, and the value r_2 determines whether the movement is toward or away from the destination. The parameter adds a random weight to the destination in order to stochastically accentuate and deemphasize the role of destination in defining the distance when r_3 . Finally, in Equation 11, r_4 shifts equally between the sine and cosine components.

4 Result Analysis

The performance of the proposed optimization algorithm is evaluated by testing on the IEEE 14-bus, 30-bus, 39-bus, 57-bus, and 118-bus test systems under five cases-

1) With and without considering Zero Injection 2) Normal condition 3) Single PMU failure 4) Single line outage 5) channel limit up to five using MATLAB software.

Figure 1 shows the implementation of the sine cosine method in the PMU placement problem. The IEEE standard test system specifications are shown in Table 1. For IEEE 14-bus, 30-bus, 39-bus, 57-bus, and 118-bus test systems, Table 2-7 illustrates the optimum PMU number and position, redundancy, installation cost, execution time in normal mode, single line outage, and single PMU loss contingency without and with ZIB, as well as various channel limits. Table.8 compares the proposed SCA to the existing PSO and MILP [13].

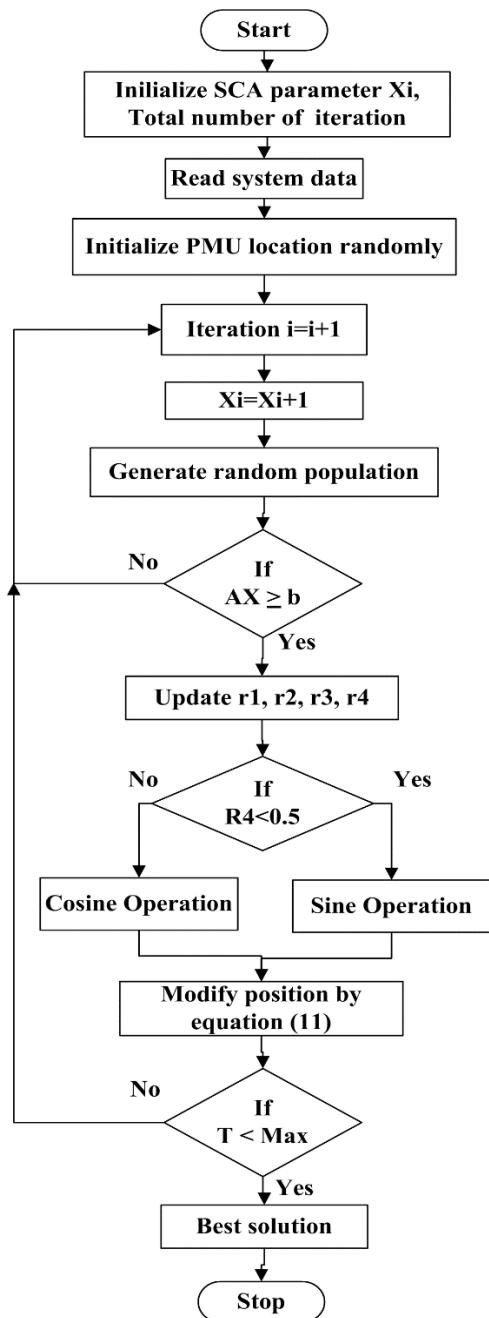


Fig.1. Flowchart for proposed algorithm in OPP problem

Table 1. IEEE standard systems specifications

IEEE System	Number of ZIBs	Location of ZIBs	No.of radial bus	Location of radial bus
14 Bus	1	7	1	8
30 Bus	6	6,9,22,25,27,28	3	11,13,26
39 Bus	12	1,2,5,6,9,10,11,13,14, 17,19,22	9	30,31,32,33,34,35, 36,37,38
57 Bus	15	4,7,11,21,22,24,26,34, 36,37,39,40,45,46,48	1	33
118 Bus	10	5,9,30,37,38,63,64,68, 71,81	7	10,73,87,111,112, 116,117

Table 2. Number of PMU and locations without considering ZIB.

Test System	Type	No. of PMU	Location of PMU	Redundancy	% between min.no. to total no. of PMUs	Installation Cost (unit)	Execution time(s)
14-bus	Normal	4	2,6,7,9	19	28.57%	3.62	0.03
	Single line outage	8	2,4,5,6,7,9,11, 13	37	57.14%	7.19	0.04
	Single PMU loss	9	2,4,5,6,9,10,13	39	64.28%	8.22	0.03
30-bus	Normal	10	1,6,7,9,10,12, 15,19,25,27	48	33.33%	9.3	0.05
	Single line outage	18	1,2,4,5,6,9,10,1 2,15,17,19,20, 22,24,25,27,28, 29	79	60%	17.3	0.05
	Single PMU loss	21	1,2,4,5,6,9,10,1 2,13,15,16,18, 20,22,24,25,26, 27,28,29	85	70%	20.17	0.01
39-bus	Normal	13	2,6,9,10,13,14,1 7,19,20,22,23, 25,29	52	33.33%	12.7	0.03
	Single line outage	19	2,3,6,8,9,10,11, 13,14,16,17,19, 20,22,23,25,26, 29,39	78	48.71%	17.9	0.04
	Single PMU loss	28	2,3,6,8,9,10,11, 13,14,16,17,19, 20,22,23,25,26, 29,30,31,32,33, 34,35,36,37,38, 39	96	71.79%	27.1	0.03
57-bus	Normal	17	1,4,6,9,15,20,24 .25,28,32,36,38, 41,46,50,53,57	72	29.82%	15.4	0.05
	Single line outage	32	1,3,4,6,9,11,12, 15,19,20,22,24, 25,26,28,29,30, 32,34,36,37,38,	128	56.14%	30.7	0.03

				39,41,44,46,47, 50,51,53,54,56			
	Single PMU loss	33		1,3,4,6,9,11,12, 15,19,20,22,24, 25,27,28,29,30, 32,33,35,36,37, 38,39,41,44,46, 47,50,51,53,54, 56	127	57.89%	31.8
118-bus	Normal	32		2,5,10,12,15,17, 21,25,29,34,37, 41,45,49,53,56, 62,64,72,73,75, 77,80,85,87,91, 94,101,105,110, 114,116	151	27.11%	31.6
	Single line outage	62		1,2,5,7,9,10,11, 12,15,17,19,21, 22,25,26,28,29, 34,35,37,41,42, 43,45,46,49,50, 52,53,56,59,62, 63,65,67,,68,70, 71,72,75,76,77, 79,80,84,85,87, 89,91,92,94,96, 100,101,105,10 7,109,110,113,1 14,115	276	52.54%	62.6
	Single PMU loss	68		1,2,5,7,9,10,11, 12,15,17,19,21, 22,25,26,28,29,, 32,34,35,37,40, 41,44,45,46,49, 52,53,56,57,58, 59,62,63,65,67,, 68,70,71,72,75, 76,77,79,80,84, 85,86,87,89,92, 94,96,100,101,1 05,107,109,110, 111,112,114,11 5,116,117,118	309	57.62%	67.1

Table 3. Number of PMU and locations for IEEE 14-bus with considering ZIB

Chann el limit	Type	No. of PMU	Location of PMU	Redun dancy	% between min.no. to total no. of PMUs	Install ation Cost	Executio n time(s)
-	Normal	3	2,6,9	55	21.43%	2.857	0.04
	Single line outage	7	2,4,5,6,9, 11,13	33	50%	6.5	0.039
	Single PMU loss	7	2,4,5,6,9, 10,13	33	50%	6.5	0.043

1	Normal	7	1,2,3,4, 10,12,13	19	50%	7.8	0.045
2	Single line outage	13	1,2,3,4,5, 6,8,9,10, 11,12,13, 14	37	92%	14.1	0.04
	Single PMU loss	12	1,2,3,5,6, 7,9,10,11 ,12,13,14	33	85.71%	14.1	0.06
	Normal	5	2,3,4,11, 13	22	35.71%	5.9	0.1
3	Single line outage	9	2,4,5,6,7, 9,11,13, 14	26	64.28%	10.5	0.12
	Single PMU loss	9	2,4,5,6,7, 9,11,13, 14	26	64.28%	10.5	0.12
	Normal	4	2,2,6,9	16	28.57%	5.1	0.21
4	Single line outage	7	2,4,5,6,9, 10,13	25	50%	9	0.19
	Single PMU loss	7	2,2,5,6,9, 10,13	28	50%	9	0.18
	Normal	4	2,2,6,9	16	28.57%	4.2	0.36
5	Single line outage	7	2,4,5,6,9, 11,13	26	50%	9.6	0.34
	Single PMU loss	7	2,2,2,6,9, 10,13	27	50%	9.6	0.36
	Normal	3	2,6,9	15	21.43%	4.5	0.52
6	Single line outage	7	1,2,3,6,9, 10,13	32	50%	10.5	0.54
	Single PMU loss	7	1,2,3,6,9, 10,13	33	50%	10.5	0.56

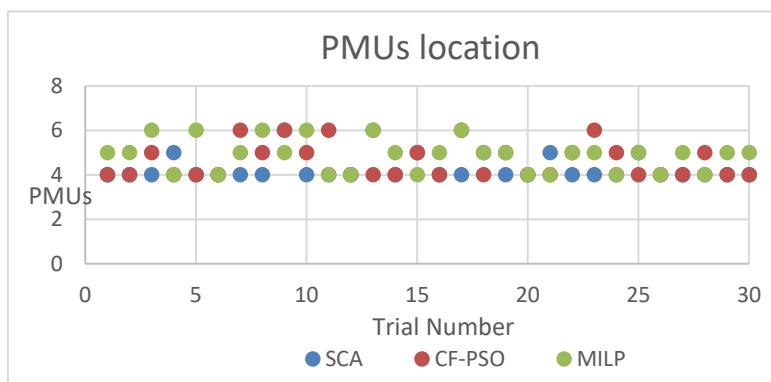


Fig 2. PMU placement in 30 trials - IEEE14 bus

Table 4. Number of PMU and locations for IEEE 30-bus considering ZIB

Channel limit	Type	No. of PMU	Location of PMU	Redundancy	Installation Cost	Execution time(s)
1	Normal	6	2,4,10,12,15,19	31	5.8	0.05
	Single line outage	12	1,2,4,5,6,10,12,15,17,19,20,24	55	11.8	0.03
	Single PMU loss	13	2,3,4,6,7,10,12,13,15,16,19,20,24	57	12.8	0.1
	Normal	12	1,2,2,2,3,4,12,15,17,18,20,24	48	12.9	0.1
	Single line outage	23	1,2,2,2,3,4,7,10,12,13,14,15,16,17,18,19,20,23,24,28,29,29,30	84	24.8	0.11
	Single PMU loss	21	1,2,3,5,6,8,10,12,13,14,15,16,17,18,19,20,22,23,24,27,29	88	26.2	0.14
	Normal	8	2,2,4,12,15,17,20,24	32	9.3	0.18
	Single line outage	15	1,2,4,5,6,7,10,12,13,15,17,18,20,22,24	56	18	0.24
	Single PMU loss	14	1,2,4,5,6,10,12,12,15,17,18,19,24,27	54	20	0.3
2	Normal	7	1,2,10,11,12,18,24	24	8.8	0.3
	Single line outage	13	1,2,4,7,10,12,13,15,17,18,20,22,24	48	16.4	0.3
	Single PMU loss	13	1,2,3,7,10,10,12,12,15,18,19,24,27	48	17.5	0.35
3	Normal	6	2,4,10,12,15,19	25	8.1	0.51
	Single line outage	12	1,2,3,5,10,12,13,15,16,18,19,24,27	48	15.9	0.51
	Single PMU loss	12	1,2,3,5,10,12,13,15,16,18,19,24,27	51	16.9	0.4
4	Normal	6	1,5,10,12,19,24	26	8.6	0.71
	Single line outage	11	1,2,4,7,10,12,15,17,19,20,24	48	15.7	0.75
	Single PMU loss	12	1,2,3,5,10,12,13,15,16,18,19,24,27	50	18	0.64

Table 5. Number of PMU and locations for IEEE 39-bus considering ZIB

No. of channel limit	Type	No. of PMU	Location of PMU	Redundancy	Installation Cost	Execution time(s)
Without channel limit						
1	Normal	7	2,3,16,20,23,25,29	31	6.8	0.03
	Single line outage	12	2,6,13,14,16,19,20,22,23,25,26,29	52	11.8	0.03
	Single PMU loss	16	2,3,6,14,15,16,20,21,23,25,26,29,34,36,37,38	58	15.8	0.06
	Normal	14	1,3,4,5,6,18,19,20,23,23,24,25,28,29	52	15.1	0.2
	Single line outage	27	1,1,2,4,4,5,5,6,6,7,16,16,17,19,20,22,23,23,24	54	25.1	0.15

			,25,26,26,28,29,34,36, 37,38			
	Single PMU loss	27	14,14,14,15,19,20,20,2 3,23,24,25,25,26,27,28 ,28,30,31,32,33,34,35, 36,37,38,39	54	29	0.14
2	Normal	9	4,4,9,20,21,25,27,29	34	10.5	0.36
	Single line outage	15	2,3,6,14,15,16,20,21,2 3,25,26,29,34,36,38	54	18	0.35
	Single PMU loss	19	1,3,4,4,5,5,16,17,20,22 ,23,23,25,26,29,29,34, 37	57	22	0.38
3	Normal	8	3,4,15,16,20,23,25,29	31	10.1	0.6
	Single line outage	12	2,6,8,15,16,17,20,23,2 5,26,29,35	51	15.2	0.52
	Single PMU loss	16	2,5,6,6,16,16,16,20,23, 25,26,29,34,36,37,38	55	18.3	0.5
4	Normal	7	3,6,16,20,23,25,29	29	9.2	0.75
	Single line outage	12	2,6,8,15,16,17,20,23,2 5,26,29,35	52	15.9	0.8
	Single PMU loss	16	2,5,6,14,16,17,20,22,2 3,25,26,29,34,36,37,38	62	21.3	1.01
5	Normal	7	6,16,20,23,25,29,39	31	10.1	1.02
	Single line outage	12	2,6,8,15,16,17,20,23,2 5,26,29,35	52	17.4	1.06
	Single PMU loss	16	2,5,6,8,16,19,20,21,23, 25,26,29,34,36,37,38	60	23.7	1.27

Table 6. Number of PMU and locations for IEEE 57-bus considering ZIB

Chann el limit	Type	No. of PMU	Location of PMU	Red unda ncy	Install ation Cost	Executi on time(s)
1	Normal	11	1,6,9,19,29,30,32,38,51,54,56	48	10.8	0.04
	Single line outage	21	1,3,4,9,12,15,19,20,25,28,29,31 ,32,38,42,48,50,51,53,54,56	86	20.8	0.07
	Single PMU loss	22	1,3,4,9,12,15,19,20,25,27,29,30 ,32,33,38,41,49,50,53,54,56	88	21.8	0.3
	Normal	21	3,3,4,9,10,11,15,16,17,19,23,24 ,28,31,33,34,47,50,53,54,56	46	22.6	0.4
	Single line outage	22	1,3,4,9,12,15,19,20,25,27,29,31 ,32,33,37,38,42,50,51,53,54,56	88	30.2	0.25
	Single PMU loss	42	2,3,3,8,9,10,12,16,16,17,18,19, 20,23,28,29,30,31,32,33,38, 41,41,43,43,45,45,48,48,49,49, 51,52,53,54,55,56,56,56,57,57	84	43.6	0.5
	Normal	14	1,3,15,20,25,29,29,32,37,41,49, 54	41	16.3	0.65
	Single line outage	23	1,1,6,9,12,12,18,20,25,27,29,30 ,32,32,36,38,41,46,50,51,53,54, 56	82	28.1	0.67
	Single PMU loss	23	1,1,6,9,12,12,18,20,25,27,29,30 ,32,32,36,38,41,46,50,51,53,54, 56	82	28.1	0.67
3	Normal	12	1,4,10,15,20,25,29,32,38,41,49, 54	45	15.3	0.96

	Single line outage	22	1,3,7,10,12,15,19,20,25,28,29,30,32,33,38,41,49,50,52,54,55,56	84	28.6	0.99
4	Single PMU loss	22	1,1,4,9,12,18,20,25,27,29,30,32,,33,36,38,41,46,50,51,53,54,56	86	28.6	0.99
	Normal	11	1,4,10,18,25,29,32,38,41,49,54	45	14.9	1.06
	Single line outage	21	1,3,6,9,10,12,15,19,20,25,27,29,,31,32,38,41,49,50,53,54,56	84	28.5	1.02
5	Single PMU loss	22	1,2,4,9,12,18,20,25,27,29,30,32,,33,37,38,41,46,50,51,53,54,56	86	29.8	1.1
	Normal	11	1,4,10,20,25,29,32,38,41,49,54	47	16.5	1.26
	Single line outage	21	1,3,6,9,10,12,15,19,20,25,27,29,,31,32,38,41,49,50,53,54,56	87	30.9	1.3
	Single PMU loss	22	1,2,4,9,12,18,20,25,27,29,30,32,,36,38,41,46,50,51,53,54,56	89	32.3	1.33

Table 7. Number of PMU and locations for IEEE 118-bus considering ZIB

Cha nnel limit	Type	No. of PMU	Location of PMU	Red unda ncy	Install ation Cost	Execut ion time(s)
1	Normal	26	2,11,12,17,21,23,28,34,40,45,49,52,56,62,71,75,77,80,85,87,90,94,102,105,110,115	257	34.2	1.7
	Single line outage	53	2,3,6,11,12,15,17,19,20,21,23,27,28,,31,32,34,35,40,42,44,45,46,49,51,5,2,54,56,57,59,62,66,70,71,75,77,78,80,83,85,86,89,90,92,94,96,100,101,,105,106,109,110,115,118	260	36.6	0.15
	Single PMU loss	57	1,3,7,11,12,15,17,19,21,22,23,27,28,,31,32,34,36,40,42,43,45,46,49,50,5,1,52,53,56,59,62,66,70,71,75,77,79,80,84,85,86,87,89,90,92,94,96,100,102,105,107,109,110,111,112,115,117,118	265	36.5	1.5
	Normal	55	1,3,4,4,7,12,13,15,16,18,21,23,24,26,28,31,32,34,35,37,41,42,43,45,47,48,50,51,54,56,59,59,62,67,72,74,76,77,7,8,80,80,84,87,88,90,91,94,95,98,10,1,103,106,107,108,110,111,115	260	58.3	0.7
	Single line outage	103	1,2,3,6,7,9,11,12,13,14,15,16,17,18,19,20,21,22,23,25,26,27,28,29,31,32,34,35,37,41,42,43,44,45,46,48,49,49,50,51,52,53,55,55,56,57,58,60,62,64,65,66,67,68,69,70,71,74,75,76,77,78,79,80,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,105,106,107,,108,109,110,111,112,113,114,115,117,118	212	112.8	0.97
	Single PMU loss	109	1,2,6,7,11,12,13,14,15,16,17,17,19,19,21,22,23,24,25,25,27,29,29,32,36,36,41,42,43,44,44,46,47,47,48,49,50,51,52,53,55,55,56,56,57,58,60,62,65,65,66,67,68,68,68,68,69,72,72,73,74,75,75,76,77,78,79,80,80,	218	118.1	0.71

				81,81,83,83,85,85,86,87,88,89,90, 92,92,94,94,95,97,97,98,99,100,101 ,102,103,104,106,106,107,108,109, 110,110,111,112,113,113,114,115, 116,117,118			
2	Normal	36		1,6,12,15,17,18,21,25,29,36,40,40,4 4,46,51,54,57,62,64,65,69,72,75,77, 80,80,82,85,86,90,94,100,102,105, 109,110,114	142	43.2	0.9
	Single line outage	57		1,3,11,12,12,13,17,18,20,21,23,27, 28,29,30,32,34,36,40,41,43,45,46, 49,52,53,56,57,58,59,62,67,70,71,7 5,77,78,80,84,85,86,87,89,90,92,94, 96,100,101,105,105,107,109,110, 111,112,115,118	211	64.2	0.75
	Single PMU loss	69		1,2,6,8,11,12,12,15,15,17,19,20,22, 23,27,28,29,32,32,34,35,41,42,44,4 5,46,49,51,52,54,54,56,57,61,62,62, 65,69,70,72,75,77,78,80,80,80,82,8 3,85,86,87,89,90,92,92,94,96,100,1 00,102,105,105,106,109,110,110, 110,115,118	209	80.6	0.82
3	Normal	29		1,11,12,15,17,21,27,31,32,34,40,45, 49,52,56,59,66,70,75,77,80,85,86, 90,92,96,100,105,110	110	37.3	1.5
	Single line outage	57		1,3,11,12,12,13,17,18,20,21,23,27,2 8,29,30,32,34,36,40,41,43,45,46,49, 52,53,56,57,58,59,62,67,70,71,75, 77,78,80,84,85,86,87,89,90,92,94,9 6,100,101,105,105,107,109,110,111 ,112,115,118	217	73.3	1.7
	Single PMU loss	58		1,2,8,11,12,12,15,17,17,20,21,23,27 ,29,31,32,34,34,40,41,43,45,46,49, 49,52,53,54,56,56,62,62,65,72,75,7 5,77,77,80,80,83,85,86,87,89,90,92, 94,96,100,100,101,105,105,110,110 ,110,115	221	72.8	1.82
4	Normal	27		1,12,15,17,21,24,26,29,34,40,45,49, 53,56,62,75,77,80,85,86,90,94,102, 105,110,114,115	111	37.6	2.09
	Single line outage	53		1,3,6,11,12,15,17,19,22,24,26,27,29 ,31,32,34,36,40,42,43,45,46,49,50,5 1,52,54,56,59,62,66,70,75,76,77,79, 80,83,85,86,89,90,92,94,96,100,101 ,105,107,109,110,114	219	80.3	3.07
	Single PMU loss	57		1,3,11,12,12,15,17,17,20,21,23,27,2 8,31,32,34,36,37,40,41,43,45,46,49, 49,52,53,56,59,62,66,70,71,75,77,7 8,80,85,85,86,87,90,91,92,94,96,10 0,102,105,105,108,110,111,112,115 ,118	230	80.6	3.05
5	Normal	26		3,12,15,17,21,24,26,28,34,40,45,49, 52,56,62,75,77,80,85,86,90,94,102, 105,110,114	120	37.8	4.72

Single line outage	53	1,3,6,11,12,15,17,19,21,22,24,26,27 ,29,31,32,34,36,40,42,43,45,46,49,5 0,51,52,54,56,59,62,66,70,75,76,77, 79,83,85,86,89,90,92,94, 96,100,101,105,107,109,110,114	233	81.7	4.85
Single PMU loss	57	1,3,11,12,12,15,17,17,20,21,23,27,2 8,31,32,34,36,37,40,41,43,45, 46,49,51,52,54,56,57,61,62,66,70,7 1,75,76,77,79,80,83,85,86,87, 89,90,92,94,96,100,102,105,105,10 7,109,110,111,112,114	243	84.3	4.51

Table.8 Optimal number of PMUs in IEEE system- Comparison with Existing Methods

Method	*C L	IEEE 14			IEEE 30			IEEE 39			IEEE 57		
		* N O	* SL L	* SP MU	* N O	* SL L	* SP MU	* N O	* SL UL	* SPM UL	* N O	* SL L	* SPM UL
SCA	-	3	7	7	6	12	13	7	12	16	11	21	22
PSO	-	3	7	7	6	12	13	7	12	16	11	21	22
MILP	-	3	7	7	6	12	13	7	12	16	11	21	22
R[17]	-	-	-	-	-	-	-	-	-	-	-	-	-
SCA	1	7	13	12	12	23	21	14	27	27	21	22	42
PSO	-	-	-	13	-	-	24	-	-	27	-	-	42
MILP	-	-	-	13	-	-	24	-	-	27	-	-	42
R[17]	-	7	13	13	12	23	24	14	23	27	21	42	42
SCA	2	5	9	9	8	15	14	9	15	19	14	23	23
PSO	-	-	-	-	-	-	-	-	-	-	-	-	-
MILP	-	-	-	-	-	-	-	-	-	-	-	-	-
R[17]	-	5	9	9	8	15	17	9	15	20	14	28	29
SCA	3	4	7	7	7	13	13	8	12	16	12	22	22
PSO	4	7	7	7	7	13	13	8	13	16	12	23	24
MILP	4	7	7	7	7	13	13	8	13	16	12	23	24
R[17]	-	4	7	7	7	13	14	8	12	17	12	23	24
SCA	4	3	7	7	6	12	12	7	12	16	11	21	22
PSO	3	7	7	6	12	12	7	12	16	11	21	22	
MILP	3	7	7	6	12	12	7	12	16	11	21	22	
R[17]	-	3	7	7	6	12	13	7	12	16	11	21	22
SCA	5	3	7	7	6	11	12	7	12	16	11	21	22
PSO	3	7	7	6	12	13	7	12	16	11	21	22	
MILP	3	7	7	6	12	13	7	12	16	11	21	22	
R[17]	-	3	7	7	6	11	13	7	12	16	11	21	22

*CL- Channel limit *N-Normal *SLO- Single line outage *SPMUL- Single PMU loss

Table 2-3 demonstrates the ideal number of PMU necessary to achieve full observability, installation cost, and execution time for IEEE 14-bus without and with ZIB along various channel limits under normal and contingency conditions. In all three cases, the ratio of necessary PMU to total number of buses decreased, as indicated in Table 2-3. At the same time, the necessary number of PMU grew when channel limitations were increased with full network observability for all optimization approaches, and then decreased as channel limits were increased.

Figure 2 shows the PMU placement in 30 trials. Figure 2 and Table 2-3 show that when SCA is combined with CF-PSO and MILP for a 14-bus system, it is more common to attain the lowest number of PMUs with the maximum measurement redundancy value.

Similarly, for IEEE 30-bus, 39-bus, 57-bus, and 118-bus test systems, Table 4-8 highlights the same observations as Table 3. When comparing the suggested optimization methodology to CF-PSO[13], MILP[13] method, and reference [17], it is found that the proposed algorithm reduces the number of PMU as the number of PMU channels increases. When compared to alternative ways, full network observability was achieved with the same number of PMU deployments. As a consequence of the findings, we can conclude that installing PMUs with more than four channels will not reduce the required number of PMUs while also increasing the installation cost. There is no need to install additional PMUs than four-channel from a cost-effective standpoint.

6 Conclusion

The study provides a sine cosine technique for achieving full network observability under normal and contingency conditions such as single line outage and single PMU loss, while taking channel availability into account. The OPP problem was validated using IEEE 14 bus, 30bus, 39-bus, 57-bus, and 118-bus test systems, and the results were compared to the existing algorithm. The suggested SCA approach outperforms then PSO and MILP and accurate for detecting PMU installation number with reduced computing time, as shown by the results. As a continuation of this work, the hybrid observability of the power system networks can be used to implement the OPP problem.

7. References

1. Mohammed, A.AI., Abido, M.: A fully adaptive PMU based fault location algorithm for series compensated lines. In: IEEE Trans. Power Syst. 29, pp. 2129-2137 (2014)
2. Georges, D.: Optimal PMU based monitoring architecture design for power systems. In Control Eng. Pract. 30, pp.150-159 (2014)
3. Banumalar, K., Manikandan B.V., Chandrasekaran, K., Arul Jeyaraj, K.: Optimal placement of phasor measurement units using clustered gravitational search algorithm. In: Journal of Intelligent & Fuzzy Systems, 34 (6), pp.: 4315-4330 (2018)
4. Saravanakumar, R., Banumalar, K., Chandrasekaran, K., Manikandan B.V.: Realistic Method for Placement of Phasor Measurement Units through Optimization Problem Formulation with Conflicting Objectives.In: Electric Power Components and Systems, 49(4-5), pp. 474-487 (2022)
5. Ebrahim Abiri, Farzan Rashidi,Taher Niknam.: An optimal PMU placement method for power system observability under various contingencies. In: International Transactions on Electrical Energy Systems, 25(4), pp. 589-606 (2015)

6. Kamyabi, L., Esmaeili, S., Koochi, M H R.: Power quality monitor placement in power systems considering channel limits and estimation error at unobservable buses using a bi-level approach, In: Int. J. Elect. Power Energy Syst., 102, pp. 302-311 (2018)
7. Shafiullah., Hossain., Abido., Fattah., Mantawy.: A modified optimal PMU placement problem formulation considering channel limits under various contingencies, In: Measurement, 135, pp. 875-885 (2019)
8. Nikolaos M. Manousakis., George N. Korres.: Optimal PMU Placement for Numerical Observability Considering Fixed Channel Capacity—A Semi definite Programming Approach, In: IEEE Transactions on Power Systems, 31(4) (2016)
9. Gou B.: Generalized integer linear programming formulation for optimal PMU placement, In: IEEE Trans Power system, 23,pp.1099-104 (2008)
10. Nikolaos C., Koutsoukis., Nikolaos M. Manousakis., Pavlos S. Georgilakis., George N. Korres. : Numerical observability method for optimal phasor measurement units placement using recursive tabu search method, In: IET Generation, Trans & Dist, 7, pp. 1- 10 (2013)
11. Marin, F.J., Garcia-Lagos, F., Joya, G., and Sandoval F. : Genetic algorithms for optimal placement of phasor measurement units in electric networks, In: Electron. Lett, 39(19), pp. 1403-1405 (2003)
12. Arul jeyaraj, K., Rajasekaran, V., Nandhakumar, S.K., K. Chandrasekaran, K.: A Multi-objective Placement of Phasor Measurement Units Considering Observability and Measurement Redundancy using Firefly Algorithm, In: J ElectrEng Technol, 10(2), pp.474-486 (2005)
13. Abiri, E., Rashidi, F., Niknam, T., Salehi, M.R. :Optimal PMU placement method for complete topological observability of power system under various contingencies, In: Electrical Power & Energy Systems, 61, pp. 585-593 (2014)
14. Chandrasekaran, K., Sankar, S., Banumalar, K. : Partial shading detection for PV arrays in a maximum power tracking system using the sine-cosine algorithm, In: Energy for Sustainable Development, 55, pp. 105-121, (2021)
15. Rahman, N.H.A., Zobaa, A.F. : Integrated Mutation Strategy with Modified Binary PSO Algorithm for Optimal PMUs Placement, In: IEEE Transactions on Industrial Informatics, 13(6), pp. 3124-3133 (2017)
16. Seyedali Mirjalili.: SCA: A Sine Cosine Algorithm for solving optimization problems, In: Knowledge-Based Systems, 96, pp.120-133 (2016)
17. Rashidi , F., Abiri,E., Niknam, T., Salehi, M.R.: Optimal placement of PMUs with limited number of channels for complete topological observability of power systems under various contingencies, In: International Journal of Electrical Power and Energy Systems, 67, pp. 125-137 (2015)
18. Chen, Joy Iong-Zong.: The evaluation of performance for a mass-mimo system with the stsk scheme over 3-d α - λ - μ fading channel, In:IRO Journal on Sustainable Wireless Systems, 1, 1-19. (2019)
19. Bashar, Abul, and Smys. S. : Physical Layer Protection Against Sensor Eavesdropper Channels in Wireless Sensor Networks, In: IRO Journal on Sustainable Wireless Systems, 3(2), pp.59-67 (2021)

Adaptive Deep Recurrent Neural Network based COVID-19 Healthcare Data Prediction for Early Risk Prediction

Asha A¹, Dhiyanesh B², *Kiruthiga G³, Shakkeera L⁴, Sharmasth Vali Y⁵, Karthick K⁶

¹Professor/ECE, Rajalakshmi Engineering College, Chennai, India.

ngash78@gmail.com

²Associate Professor/CSE, Dr. N.G.P. Institute of Technology, Coimbatore, India.

dhiyanu87@gmail.com

³Associate Professor/CSE, IES College of Engineering, Thrissur, India.

kirthikacsehod@gmail.com

⁴Associate Professor/CSE, Presidency University, Bengaluru, India.

shakkeera.1@presidencyuniversity.in

⁵Assistant Professor/CSE, Presidency University, Bengaluru, India.

sharmasth.vali@presidencyuniversity.in

⁶Assistant Professor/IT, Sona College of Technology, Salem, India.

karthickk@sonatech.ac.in

*Kiruthiga G

Abstract. The Covid-19 pandemic has spread rapidly across the globe and is now one of the leading causes of death and illness worldwide. Existing approaches for controlling coronavirus disease are challenging because, the improper solutions, medications, and data are irregular to analyze. This paper proposes Adaptive Deep Recurrent Neural Network - based Covid-19 Healthcare data prediction, where the risk prediction algorithm is made to detect the covid disease when it is typically premature. The initially collected Covid-19 sample test dataset is trained in the preprocessing step to remove irrelevant data. The margins of features are estimated using threshold values to find the defect rate based on the Intrinsic Covid Defect Rate. The trained data is processed for feature selection using a threshold value to identify the best features using Relative Cluster-Intensive Feature Selection. The selected features are introduced to an Adaptive Deep vectorized Recursive Neural Network (ADVRNN) to predict the coronavirus affected rate. The results of the proposed ADVRNN experiment improve prediction accuracy, recall, f-measure, and precision rate to enhance the early detection prediction performance compared to the existing systems.

Keywords: Covid-Prediction, Feature selection, Classification, Deep neural network, RNN, ICDR, PHR data analysis.

1 Introduction

Cloud Computing provides various services to the user based on the service infrastructure. The medical information process with data analysis provides more excellent centralized service. Because of the centralized process, information is more significant in higher Personalized Healthcare Processing (PHR). The main goal is to provide services analyzed based on feature selection and medical healthcare data analysis classification. When the evaluation index of the Covid-19 epidemic was constructed using multi-source data, the old district's risk level was much higher than in the new community. Population density is the most critical determinant of infectious diseases.

To carefully evaluate the disease prediction used for early detection and forecasting based on providing the dataset, increasing the quick development report and analysis, and providing treatment is the finest and most effective way to inhibit the production of a new type of coronavirus pneumonia. Presently, clinical data maintenance is a significant issue in the clinical field because of time series data in highly dependable various differential feature evaluation. Healthcare should offer quality support of patient data covering illness-related information and predictive analysis. So clinical Data Mining is to such an extent that puts away information that can't contain missing qualities and excess information. Enormous information creation and information reduction are fundamental before applying an information handling calculation that can influence detection results. Precise and predictable error-free information makes fast and simple diagnosis.

A new neural network structure is proposed for Covid-19 detection based on feature extraction and function-based confidence classification module rotation network data. Machine Learning (ML) is the science of training a machine that uses a mathematical model to analyze the learned data. ML has been implemented to analyze the data and detect interest patterns. Then, verified data are classified following the learning patterns during the learning process. Feature Selection in data processing technology that minimizes data generation begins to function efficiently with dimensionality reduction. The Recurrent Neural Networks (RNNs) display an exquisite method of providing regular health information. Nonetheless, one disadvantage of RNNs is that forecast execution diminishes when the line length is excessively high. To conquer this deficiency, two-way Bidirectional Recurrent Neural Networks (BRNNs) are assigned in this proposed model as an adaptive model that can be prepared to utilize all available users' data from the two headings to improve figure execution. RNNs refer to any recurrent neural network with an activation function to predict the class.

2 Related works

The survey aims to provide state-of-the-art methods for introducing researchers' information, which will clarify how Machine Learning and Deep Learning, as well as data, enhance the Covid-19 state in a broader health community. It delves into

considerable details about the obstacles and the way forward.

R. Nandakumar et al. [1], described that the modern Covid-19 mathematical models, such as box models, statistical models, and machine learning models, help to understand which models are better suited for disease prevalence analysis. V. Z. Marmarelis et al. [2], proposed a new Adaptive Phase-Space Approach (APSA) method based on a data-based detection layer and contagious wave layer. Each of these layers was described by the Riccati equation with adaptive estimated parameters.

H. Gao et al. [3], developed Mean Field Evolutionary Dynamics (MFED). It was influenced by graph mean field games and optimum transport theory, which were employed in MFED to regulate the evolution of fads by deriving multiple individual state gain functions from frequently used replication dynamics. A. M.K [4], proposed a mathematical model for describing the Covid-19 disease and predicting future waves of the illness. Forecasting was crucial for the health system's readiness and the action plan to be carried out. It was suggested that the Covid-19 disease be described using Gaussian mixture models.

O. Tutsoy et al. [5], introduced a new comprehensive, high-order, multidimensional, robust correlative, and parametric Suspected Infectious Disease mortality model. A mathematical analysis of Turkey's death toll shows that the dynamics of Covid-19 fluctuate little in a stable (limited) state. However, some dynamics are near the unstable region (infinity). G. E. Alvarez et al. [6], proposed the mathematical model for analyzing the behavior of Argentina's power system when considering the impact of Covid-19 on the population. The model achieved an accurate solution achievable in a short computation time.

H. Friji et al. [7], proposed that the generalized dynamic model with eight states to describe the progression of the Covid-19 epidemic from vulnerable to displaced conditions through isolation and hospitalization. Model parameters were determined using the three observable inputs to solve an appropriate optimization problem (infections, deaths, and reported cases). The work of F. Riquelme et al. [8], were particularly interested in the various datasets and epidemiological models used. The search strategy included four combined searches on Google Scholar from January 2020 to January 2021. The results showed 30 data sources and 11 marine and terrestrial data sources used to collect air travel.

The report by Kumari et al. [9], summarized the comprehensive review of newly developed predictive models and predictions for the number of Covid-19 cases all over the country, including confirmed, recovered, and fatal cases. Multiple linear regressions, autocorrelations, and correlation coefficients were employed for improving accuracy and predictive ability. An efficient method by using an unsupervised Deep Generative Learning-based 1-SVM (DGL-ISVM) data-driven technique to detect Covid-19 infection from blood test results was presented by A. Dairi et al. [10]. By applying an unsupervised deep mixed model to a blood test, Covid-19 infection was detected.

A model using Ordinary Differential Equations was described by Giamberardino et al. [11] as being specifically developed to describe Covid-19's evolution in Italy. In the case of an Italian population distribution model based on

national data, obtaining a numerical solution effectively reproduces the accurate data. F. Rustam et al. [12], demonstrated that the ML model could forecast the future number of patients infected with Covid-19. Specifically, the study used four standard predictive methods: Support Vector Machine (SVM), Least Absolute Sum, Linear Regression, Exponential Smoothing, and Selection Operator to measure the predictability of the Covid-19 factor.

R. F. Sear et al. [13], explained that Machine learning measures the Covid-19 content of online institutional health guidance, especially for those who oppose vaccines (anti-vaccination). The Covid-19 debate focused on the pro-vaccine (pro-vax) community rather than the anti-vaccine community. Rahman et al. [14] illustrated that the Covid-19 lockdowns significantly improved air quality and reduced respiratory conditions and illnesses as a result of Covid-19. For solving complex and intractable problems, like global pandemics, machine learning proved to be a powerful, convenient, and robust analytical paradigm.

R. F. Albuquerque Paiva de Oliveira et al. [15], assessed the correlation between laboratory parameters and Covid-19 test results. Two classification models were developed: the first for the test results in covid-19 patients and the second for test parameters in hospitalized patients for cell classification. C. Zhan et al. [16], proposed that the ML method for Covid-19 forecasting was based on Broad Learning System (BLS). Random Forest (RF) was used to show salient features. An RF-packed BLS method was developed combining packing strategies and BLS, to forecast the course of the Covid-19 epidemic.

D. C. d. S. Gomes et al. [17] described an interval type 2 fuzzy clustering algorithm with adaptive similarity distance mechanisms. A relationship between behavior and epidemiological uncertainty appears to be the mechanism of interval type 2. Observer/Kalman Filter Identification and real-time prediction were incorporated into an ambiguous version of adaptive monitoring based on unobservable components determined by recursive spectrum decomposition of experimental, epidemiological data. P. Wu et al. [18], proposed a framework for optimizing RF and SVM models, primarily using the Slim Mold algorithm (SMA). Training optimal SVM and RF models based on SMA to identify critical factors was conducted. Comparative experiments were conducted using RF-SMA and some well-known ML techniques based on Covid-19 data.

According to G. S. Dlamini et al. [19], whole-genome sequencing data from eight pathogenic strains, including SARS-CoV-2, were analyzed for Endogenous dinucleotide gene signatures. By using the extreme gradient boosting model (XGBoost), DNA sequences were converted into biased dinucleotide frequencies. E. Capiansoghi et al. [20] developed a computerized system to extract clinical, radiological, and laboratory variables pertinent to patient risk prediction, which was analyzed by clinicians when assessing the patient's risk. Using simple decision criteria, the system was intended to produce an interpretable machine learning system.

1.1 Problem Identification Factors

- The main problem is that improper feature selection leads to high dimension invariant scaling level, hence the classification doesn't produce the best performance.
- Due to learning progress weights, lower precision and recall rate, it is irregular to train in the neural network.
- Irrelevant grouping of clusters doesn't mean the absolute threshold margins to choose the features.
- The false rate and time complexity are increased due to the back rotation threshold values. So, the low prediction accuracy causes failure of the Covid ratio.

3. Proposed Method

The proposed method to detect Covid is based on data analysis on an adaptive deep learning model. It aims to contribute to early risk prediction by analyzing the feature to classify the risk. In the proposed method, an Adaptive Deep vectorized Recursive Neural Network (ADVRNN) efficiently handles the large dataset values and improves the performance accuracy and prediction results without complexity.

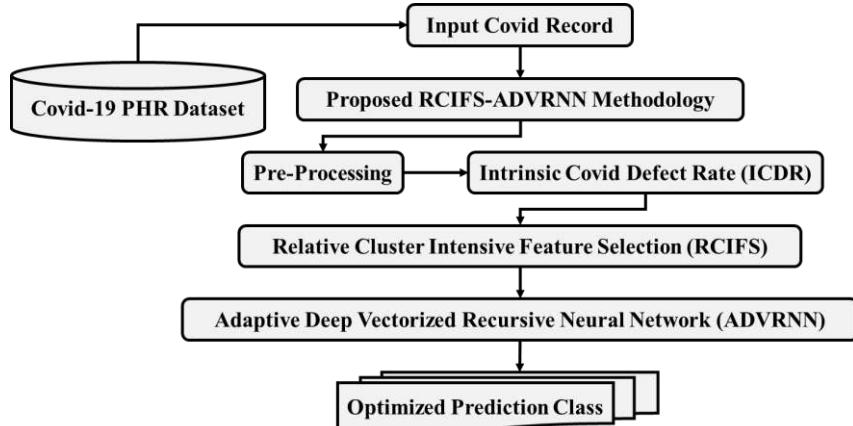


Fig. 1. Proposed Architecture RCIFS-ADVRNN

An Adaptive deep Recurrent Neural Network based Covid-19 Healthcare data prediction for early risk detection has been designed to predict the covid ratio. The significance of information determination depends on a streamlined set from the collected dataset from PHR records. It is critical to highlight the decision ratio from medical margins and choose the time series that are regularly drawn nearer to increase or decrease margins from the scaling rate, to develop the feature selection and classification based on the optimized neural variety in deep models.

The proposed method has been trained to detect Covid-19 infections more quickly by examining defect levels of relative feature margins to get closer to improving feature selection and classification accuracy. The implementation defines the ADVRNN algorithm and trains the testing and training on the medical margins having the intrinsic rate depending on n Intrinsic Covid Defect Rate (ICDR) level to select the feature. Then the evaluation is in ICDR to choose the threshold margins. A further preliminary processed dataset is trained into the feature selection process to identify the best features of Covid-19 using Relative Cluster-Intensive Feature Selection (RCIFS). The selected features get introduced to an ADVRNN.

3.1 Preprocessing

The augmented reality of the dataset was first processed with dataset preparation, as noise and feature edges are checked at this stage. This support confirms the medical margins and the presence of the dataset. Initially, the dataset is imported to identify the missing values and irrelevant data to categorize in the dataset. Then the feature scaling or selection process is performed.

Algorithm

```

Input: Initialize Covid-19 dataset- Cds
Output: Preprocessed dataset (Cov-Pd)
Step 1: Initialize to read X→{Cds1, Cds2,.....}
Step 2: Process X(I, J) for all read data to check fields (Feature)
Step 3: Index Feature counts and List Fl
        Check id Null and empty for all records
        Compute If Fl← X(I,j) null free
        Reorder index Fl for all X
        Check margins and scaling range of PHR obtained from attributes
        End For
Step 4: Return Fl←Ordered sling verified index
Step 5: Cov-Pd(Fl)

```

The algorithm returns the preprocessed values of feature sets from the PHR dataset. Here, Cds- Covid-19 dataset, Cov-Pd- Preprocessed dataset, X –Read the weight, I,j- Dataset feature values, and Fl– Feature Index which is indexed to get the feature limits and verification of scaling levels presented and described in medical margins. It returns the noise free dataset for identifying the Covid defect rate.

3.2 Intrinsic Covid Defect Rate

In this phase, Covid infection dependencies were estimated from the preprocessed Covid-19 dataset. This identifies the current margin and time series deficiency, increasing rate of the feature margins, and their variations. Also, this estimate of the interrelationship coefficient is of variation scaling of feature limits and the feature coefficient that determines the defect affected range. This selects the feature margins for maximum subset of features, combined with the defect rate coordinated feature limits for feature evaluation.

Algorithm for ICDR

```

Input: Preprocessed dataset Cov-Pd (Fl)
Output: Covid defect rate (Cdr)
Begin
Step 1: Process the index records Cdr
Step 2: Estimate the Feature margins (Fm) for all records
        Select max scale medical feature (Mf) limits (F.I.)
        Compare the margins (Mf) and (F.I.)
        FI ← (Mf) Max (F.I.)
        Evaluate Ec distance for al Fl
        End For
        Return different scale Choose max limits (Fl (Maxl))
Step 3: If max weight sustains to Compute the Coordinated feature deficiency, Cfi
        Repeat Fi
        Process For each Fi(Cfi)
        Compute Feature margin Level ==medical margin
        Find the coordinated feature cross limit( $m_f$ )
        End for
        Select subset index weight W(R) of feature relation average
        For R=1 to choose mean weight W(R)
        
$$W(R) = W(R) - \sum_{a=1}^n diff(R, Ws, D) / (mXI)$$

        
$$+ \sum_{R \neq class(F)} \left[ \frac{m(R)}{1-m(class(F))} \right] \sum_{a=1}^n diff(R, Ws, D) / (mXI)$$

        End For
Step 4: Estimate the absolute integrated mean rate (Amir)
        For each feature (( $m_f$ 1 U ( $m_f$ 2)))
        Return( $Amir_f$ ) ← W(.R.)
        End for
Step 5: Select the maximum deficiency for each class Cdr ← ( $Amir_f$ )
End if

```

The algorithm above provides the feature deficit rate obtained by sub tapping the feature boundaries. Here, Cov-Pd (Fl) - Preprocessed dataset, Fm- Feature margins, m_f -Medical feature, R- Relation average, AMIR - Absolute Integrated Mean Rate, and Ec - Euclidean distance which is the difference between two values limits, and it reduces the unaffected index edges compared to the medical thresholds to improve

the feature score. This way, the deficit level is categorized by class based on the deficit margins.

3.3 Relative Cluster Intensive Feature Selection

This relative feature has been selected based on the threshold definition attained by medical margins. This chooses the coordinated scaled value by the disease relevance y , measured based on the fitness evaluation. This determines the best fit case feature margins to scale the feature dependencies relative to the similarity level. And the margin range is estimated for coefficient scaling of feature limits, and the defect affected range determines the feature coefficient.

It groups the search dependencies, establish integrated comparisons to find similar features, and groups them into cluster indexes. The evaluation was carried out to predict the feature index by estimating the best-case similarity at an intensive level.

For choosing max scale $Mx \leftarrow F_{si}$, the best match case feature limits are chosen by getting F_{si} samples from $Cdr \rightarrow x = x_i w_{ij}$ which defines integration, where Z_{ij} is the best-case feature evaluation from each threshold margin.

$$Z_{ij} = x_i w_{ij}; Z_j = \sum_i Z_{ij} + X_j = g(Z_j) \quad (1)$$

Similar features are grouped into cluster F_{ci} as F at each feature. For choosing the relative index cluster at centroid ' r ' at the closest mean weight,

$$f(x_{fc}) = \sum_i r_i \quad (2)$$

To choose the best-case relevance feature based on kernel attention to get support value using kernel Function KL , new scaling features are obtained.

$$Class st = \frac{1}{2n} \sum_{i=1}^n (\hat{x}_i - x_i)^2 + \beta \sum_{j=1}^m KL(p|\hat{p}_j) + \frac{\lambda}{2} \sum_{i=1}^n \sum_{j=1}^m \theta_{ij}^2 \quad (3)$$

By grouping the best-case process at ' p ' class index,

$$P(F_{max \rightarrow c}) = \sum_{\omega \in A} p(\omega) \quad (4)$$

similarity in the relative margin $p(\omega) \geq 0$, as same feature levels are grouped by centroid value to return group weighted Max class.

The decision is carried out to group the relative features based on max margins to recommend the suggestive class. These are grouped into cluster index margins and scale the weights of comparable groups combined with others to form ascending index. This reduces the non-related features and increases the feature selection accuracy for further classification.

3.4 Adaptive Deep vectorized Recursive Neural Network

In this phase, feature selection cluster groups are trained into the proposed adaptive vectorization and scaling model deep recurrent neural network. The proposed ADVRNN algorithm identifies the feature limits' importance with a re-activated logical definition. It constructs a $16 * 16$ scaling iteration feed-forward layer to train the feature cluster with an analytical linearity learning model. It trains the feature

dependencies and logical decisions by activating the conditional definition with scaled medical margins based on the threshold limits. It reduces the negative impact of non-relation features to predict the risk by definition and increases accuracy.

Algorithm for Adaptive Deep vectorized Recursive Neural Network

Input: Preprocessed dataset Cov-Pd (Fl)
 Input: Feature cluster group 'St.'
 Output: Predicted class by risk
 Begin
 Initialize the relational cluster index St → Fs
 Evaluate the vector coloration based on matrix index Vmt
 Set the initial layers based on the feature weights
 Set the modified iterative RNN limitations ε, μ, β (Vmt)
 Set feature Clusters Fs(Vmt)
 Train the consequences of the feature in feed-forward In neural weights
 For $\varepsilon = 1$ cluster class, Vmt to \in do
 For each cluster class to form iteration Let accept μ refers to presentation rate, ε - Repetition phase $\varepsilon \leftarrow 0$ $(1 + a)^n = 1 + \left(\frac{na}{1!}\right) + \left(\frac{n(n-1)a^2}{2!}\right)$
 Compute the loss ranges a lr → \in -maximum number of iterations β -number of images covered iterations
 Compute progressive decision on each margining medical scale ω^* at n cluster class
 Chose the Max scale to feature weight compared to medical margin
 Select the index scale by class by preference.
 End

The ADVRNN predicted the covid defect rate based on the training class, and the margins are fitted through threshold margins. The best case measure validates this by testing the medical margin comparison to predict the class. μ refers to performance rate, ε - Iteration stage $\varepsilon \leftarrow 0$ $(1+a)^n = 1 + \left(\frac{na}{1!}\right) + \left(\frac{n(n-1)a^2}{2!}\right)$, \in -maximum number of iterations, and β -number of images covered iterations. The classification results also show the prediction accuracy based on the margin classes to point from different threshold margins depending on risk by category.

4. Experimental Results and Discussion

The implementation was simulated to prove the result performance, and the UCI database was used to test the performance of various classifiers. In this Covid-19 test, samples have 30 features to handle classification prediction. Features are (Day wise, Month wise, Patient Name, Affected date, temperature, gender, Spo2, etc.). These attributes are included in Covid-19 data samples dataset. Confusion matrix

rules are to be followed to understand the version of the model for training and testing. Table 1 shows the proposed implementation environment consideration and its values. The proposed approach is implemented under various parameters, and performance is evaluated with a Covid-19 dataset from healthcare monitoring data. This method measures efficiency in the prognosis of a disease based on multiple functions and their values. The evaluation results are compared to those of other approaches like Deep Generative Learning-based 1-SVM (DGL-ISVM), Adaptive Phase-Space Approach (APSA), and Adaptive Synthetic approach (ADASYN).

Table 1: Environment and parameters processed

Parameters	Values
Cloud Environment	AWS (Amazon web service)
Storage	EBS
Configuration	Txlarge core2
Language, Tool Used	Python, Jupiter notebook.
Dataset Used	Covid-19 test samples data

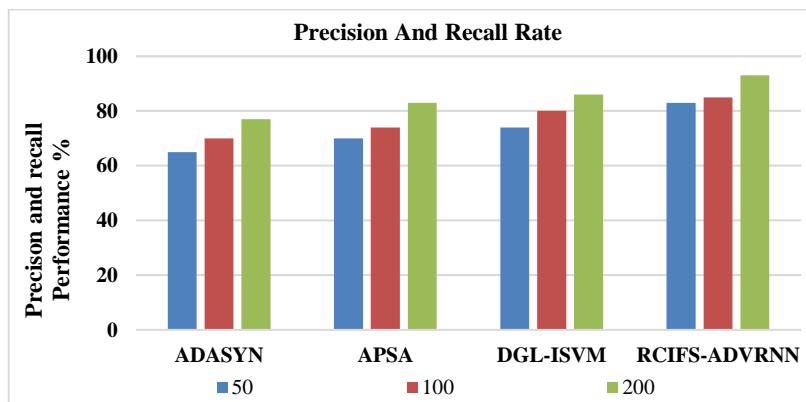


Fig. 2. Performance in Precision and Recall Rate

The performance of routing in the network is measured at different nodes with different numbers and is illustrated in Figure 2. The proposed RCIFS-ADVRNN system has a higher precision and recall rate efficiency than other methods at all levels.

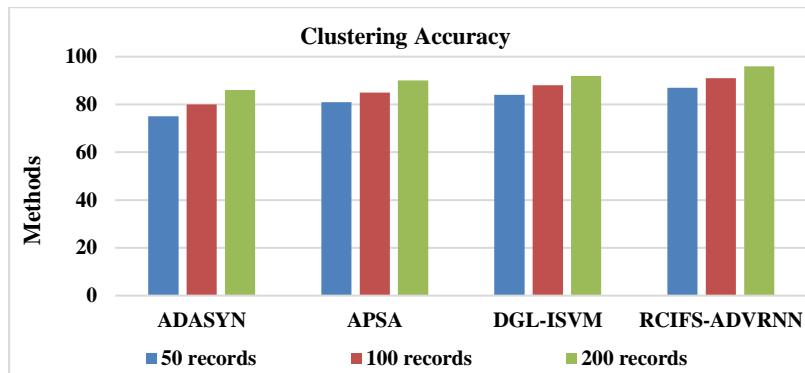


Fig. 3. Accuracy in Clustering

Fig. 3. shows the clustering performance generated by different methods. The proposed RCIFS-ADVRNN approach has developed high clustering accuracy under several other diseases.

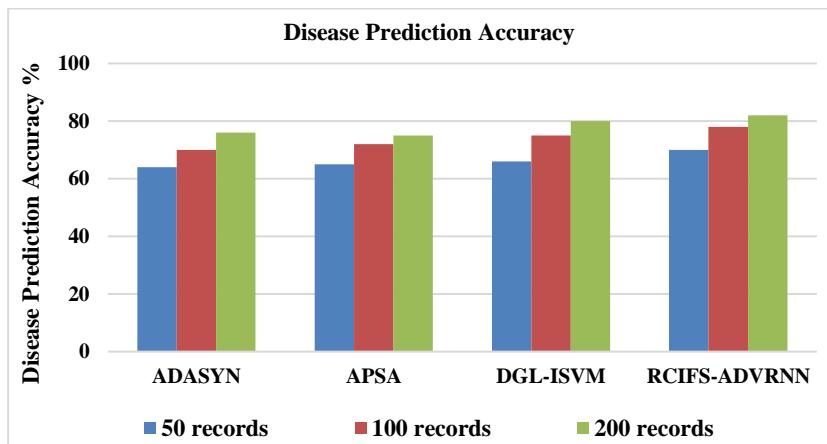


Fig. 4. Analysis of Disease Prediction Accuracy

The accuracy of the prognosis generated by the various methods was measured and is shown in Fig. 4. The proposed hybrid approach improved disease outcomes compared to other approaches in each class.

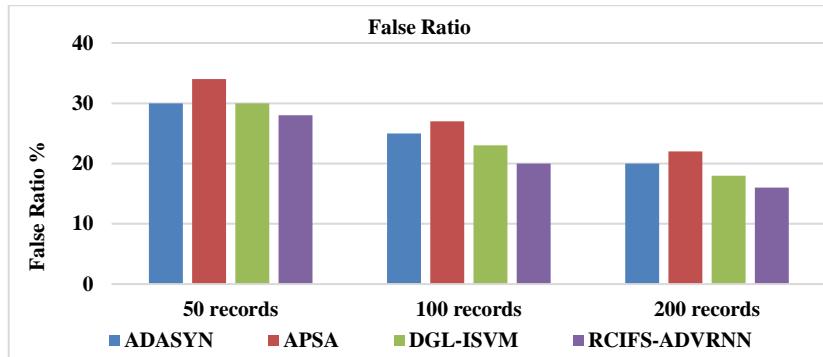


Fig. 5. Analysis of False Classification Ratio

Fig. 5 illustrates the accuracy of disease prognosis based on different techniques. A higher disease prognosis was demonstrated by the proposed RCIFS-ADVRNN approach than by the approaches from other classes.

5. Conclusion

The proposed Adaptive deep Recurrent Neural Network is based on Covid-19 healthcare data prediction for early detection. The proposed Relative Cluster Intensive Feature Selection (RCIFS) selects the critical feature to reduce the dimension ratio. The chosen elements are trained with Adaptive Deep vectorized Recursive Neural Network (ADVRNN) to accurately predict the covid affected rate based on medical threshold margins. The proposed RCIFS-ADVRNN system attains high performance compared to the existing systems. This proves high classification and feature selection accuracy with a precision rate of 97.2 %, recall rate of 97.6%, and classification cluster accuracy of 98.2 %, which are best compared to other systems. Future work will include exploring the model with a massive dataset and evaluating the model with an external dataset deep learning model to be optimized with adaptive methodologies.

References

- [1] R. Nandakumar, V. Ponnusamy, K. C. Sriharipriya, and J. C. Clement.:A Survey on Mathematical, Machine Learning and Deep Learning Models for COVID-19 Transmission and Diagnosis. IEEE Reviews in Biomedical Engineering, vol. 15 (2022)
- [2] V. Z. Marmarelis.: Predictive Modeling of Covid-19 Data in the U.S.: Adaptive Phase-Space Approach. IEEE Open Journal of Engineering in Medicine and Biology, vol. 1, (2020)
- [3] H. Gao, W. Li, M. Pan, Z. Han, and H. V. Poor.: Modeling COVID-19 with mean field evolutionary dynamics: Social distancing and seasonality. Journal of Communications and Networks, vol. 23, no. 5 (2021)

- [4] A. M.K.: Mathematical Modeling of COVID-19 and Prediction of Upcoming Wave. *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 2, (2022)
- [5] O. Tutsoy, S. Çolak, A. Polat and K. Balikci.: A Novel Parametric Model for the Prediction and Analysis of the COVID-19 Casualties. *IEEE Access*, vol. 8, (2020)
- [6] G. E. Alvarez and J. L. Sarli.: Mathematical model to control the Argentine Energy System during the COVID 19 pandemic. *IEEE Latin America Transactions*, vol. 19, no. 6, (2021)
- [7] H. Friji, R. Hamadi, H. Ghazzai, H. Besbes, and Y. Massoud.: A Generalized Mechanistic Model for Assessing and Forecasting the Spread of the COVID-19 Pandemic. *IEEE Access*, vol. 9, (2021)
- [8] F. Riquelme, A. Aguilera, and A. Inostrosa-Psjas.: Contagion Modeling and Simulation in Transport and Air Travel Networks During the COVID-19 Pandemic: A Survey. *IEEE Access*, vol. 9, (2021)
- [9] R. Kumari et al.: Analysis and predictions of spread, recovery, and death caused by COVID-19 in India. *Big Data Mining and Analytics*, vol. 4, no. 2, (2021)
- [10] A. Dairi, F. Harrow and Y. Sun.: Deep Generative Learning-Based 1-SVM Detectors for Unsupervised COVID-19 Infection Detection Using Blood Tests. *IEEE Transactions on Instrumentation and Measurement*, vol. 71, (2022)
- [11] P. D. Giamberardino, D. Iacoviello, F. Papa and C. Sinisgalli.: Dynamical Evolution of COVID-19 in Italy with an Evaluation of the Size of the Asymptomatic Infective Population. *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, (2021)
- [12] F. Rustam et al.: COVID-19 Future Forecasting Using Supervised Machine Learning Models. *IEEE Access*, vol. 8, pp. 101489-101499. (2020)
- [13] R. F. Sear et al.: Quantifying COVID-19 Content in the Online Health Opinion War Using Machine Learning. *IEEE Access*, vol. 8, pp. 91886-91893. (2020)
- [14] M. M. Rahman, K. C. Paul, M. A. Hossain, G. G. M. N. Ali, M. S. Rahman and J. -C. Thill.: Machine Learning on the COVID-19 Pandemic, Human Mobility and Air Quality: A Review. *IEEE Access*, vol. 9, (2021)
- [15] R. F. Albuquerque Paiva de Oliveira, C. J. A. Bastos Filho, A. C. A. M. V. F. de Medeiros, P. J. Buarque Lins dos Santos and D. Lopes Freire.: Machine learning applied in SARS-CoV-2 COVID 19 screening using clinical analysis parameters. *IEEE Latin America Transactions*, vol. 19, no. 6, (2021)
- [16] C. Zhan, Y. Zheng, H. Zhang, and Q. Wen.: Random-Forest-Bagging Broad Learning System with Applications for COVID-19 Pandemic. *IEEE Internet of Things Journal*, vol. 8, no. 21, (2021)
- [17] D. C. d. S. Gomes and G. L. d. O. Serra.: Machine Learning Model for Computational Tracking and Forecasting the COVID-19 Dynamic Propagation. *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 3, (2021)
- [18] P. Wu et al.: An Effective Machine Learning Approach for Identifying Non-Severe and Severe Coronavirus Disease 2019 Patients in a Rural Chinese Population: The Wenzhou Retrospective Study. *IEEE Access*, vol. 9, (2021)
- [19] G. S. Dlamini et al.: Classification of COVID-19 and Other Pathogenic Sequences: A Dinucleotide Frequency and Machine Learning Approach. *IEEE Access*, vol. 8, (2020)
- [20] E. Casiraghi et al.: Explainable Machine Learning for Early Assessment of COVID-19 Risk Prediction in Emergency Departments. *IEEE Access*, vol. 8, (2020)

Maximum Decision Support Regression Based Advance Secure Data Encrypt Transmission For Healthcare Data Sharing In The Cloud Computing

Anusuya V¹, *Bejoy B J², Ramkumar M³, Shanmugaraja P⁴, Dhiyanesh B⁵, Kiruthiga G⁶

¹Associate Professor / CSE, Ramco Institute of Technology, Rajapalayam, India.

pgkrishanu@gmail.com

²Assistant Professor/CSE, CHRIST (Deemed to be University), Bangalore, India.

2*bejoybj@gmail.com

³Associate Professor/CSBS, Knowledge Institute of Technology, Salem, India.

3hod.csbs@kiot.ac.in

⁴Associate Professor/IT, Sona College of Technology, Salem, India.

4shanmugarajap@gmail.com

⁵Associate Professor/CSE, Dr. N.G.P. Institute of Technology, Coimbatore, India.

5dhiyanu87@gmail.com

⁶Associate Professor/CSE, IES College of Engineering, Thrissur, India.

6kirthikacsehod@gmail.com

*Bejoy B J

Abstract. The recent growth of cloud computing has led to most companies storing their data in the cloud and sharing it efficiently with authorized users. Healthcare is one of the initiatives to adopt cloud computing for services. Both patients and healthcare providers need to have access to patient health information. Healthcare data must be shared and maintained more securely. While transmitting health data from sender to receiver through intermediate nodes, intruders can create falsified data at intermediate nodes. Therefore, security is a primary concern when sharing sensitive medical data. It is thus challenging to share sensitive data in the cloud because of limitations in resource availability and concerns about data privacy. Healthcare records struggle to meet the needs of security, privacy, and other regulatory constraints. To address these difficulties, this novel proposes a machine learning-based Maximum Decision Support Regression (MDSR) based advanced secure Data Encrypt Transmission (ASDET) approach for efficient data communication in cloud storage. Initially, the proposed method analyzed the node's trust, energy, delay, and mobility using Node Efficiency Hit Rate (NEHR) method. Then identify the efficient route using an Efficient Spider Optimization Scheme (ESOS) for healthcare data sharing. After that, MDSR analyzes the malicious node for efficient data transmission in the cloud. The proposed Advanced Secure Data Encrypt Transmission (ASDET) algorithm is used to encrypt the data. ASDET achieved 92% in security performance. The proposed simulation result produces better performance compared with PPDT and FAHP methods.

Keywords: Health Care, Cloud computing, Machine Learning, Security, MDSR, NEHR, ASDET, Malicious Node, Energy, Delay.

1 Introduction

Manually storing health records and retaining them for future reference is challenging in managing large amounts of data. The problem with the traditional system of storing all data manually or on paper is the difficulty of locating patient data in record rooms with large numbers of health records. Finding patient-specific medical records requires a lot of time and effort. Data is plain text and easily stolen. Therefore, anyone can easily access the data in the form to read, write or modify it. In the modern world, electronic medical records are widely stored in the cloud. Cloud computing is a growing technology that frees users from the burden of hardware maintenance and provides dynamically adaptable and scalable computing resources that can be accessed anywhere a network is available. On tight budgets, organizations and the healthcare industry can benefit from advanced computing and storage services without investing in infrastructure and maintenance.

However, the loss of control over data and computing raises several security concerns for organizations and hinders the broader applicability of the public cloud. However, data sharing in the cloud requires advanced security measures because data can easily be lost, leaked, or stolen. While transmitting health data from sender to receiver through intermediate nodes, intruders can create falsified data at intermediate nodes. Therefore, security is a primary concern when sharing sensitive medical data. Data privacy and lightweight operations on resource-constrained sensor nodes are two challenges associated with sharing sensitive data in the cloud.

Health data includes information about an individual's medical history, records, and other personal data. traindata.csv – File containing features of each case related to patient, hospital, and duration of stay traindata_dictionary.csv – File containing information about feature training files. The test set: Testdata.csv – file containing patient and hospital-related characteristics. Need to predict the length of stay for each case I.D.

1.1 The novelty of this paper

- To solve the above problems, the proposed method initially analysis the node's trust, energy, delay, and mobility using Node Efficiency Hit Rate (NEHR) method.
- Then identify the efficient route using an Efficient Spider Optimization Scheme (ESOS) for healthcare data sharing.
- After that, MDSR analyzes the malicious node for efficient data transmission in the cloud. The proposed Advanced secure Data Encrypt Transmission (ASDET) algorithm is used to encrypt the data.
- The proposed simulation result produces better performance compared to with PPDT and FAHP methods.

2 Related work

J. Liang et al. (2021), The author proposes that the efficient scheme for monitoring the healthcare systems is the Privacy Preserving Decision Tree (PPDT). First, the clinical decision tree will be changed as a Boolean vector. The PPDT is used to get by search the encrypted indices and encrypted tokens. PPDT is more effective in evaluating, transmitting, and storing performance analysis. However, this method does not produce healthcare records.

A. Jindal et al. [1], The author propose that the data classification method for efficient decision-making in fuzzy rules-based classifiers is designed. A cloud computing environment gives effective results in different performance evaluation metrics. It is very challenging to store extensive data in the dataset. X. Li et al. [2], The author describes the cloud-based medical storage system as the Efficient Privacy-Preserving Public Auditing Protocol (EPPPAP). That information is under the defined security model in detailed security analysis. The challenge is recovering the oldest data is difficult.

A. Agrawal et al. [3], It is shown in the author's article how fuzzy AHP-TOPSIS can be used to select an order of preference based on similarity to an ideal solution. Similarly, L. Wang et al. [4], the author describes the method AHP-TOPSIS. This method suggests future use of the Internet of Health Things (IoMT).

M. Zarour et al. [5], The paper presents a method for computing blockchain technology's impact and a new direction for future research. Criteria weights are calculated using the fuzzy Analytical Network Process (F-ANP). H. Abrar et al. [6], The author proposes identifying the essential assets of the Healthcare Information System (HIS) for the security purpose of cloud computing models. And the HIS impact the assessing the similarity.

S. More et al. [7], Image estimation amounts are measured using Peak Signal Noise Ratios (SNRs), Structural Similarity Indexes (SSIs), and Mean Squared Errors (MSEs). It increases the image's potential and improves the image's visualization.

Y. K. Saheed et al. [8] implemented the IoMT method in Deep Recurrent Neural Network (DRNN). The DRNN method is used to supervise machine learning methods, and then the IoMT has Random Forest, Decision tree, and Ridge Classifier. Then IoMT environment generates a powerful and well-organized IDS for DRNN.

W. Niu et al. [9], Data transmitting is reinforced, communication systems were designed rationally, and all essential data was utilized. X. Wang et al. [10] suggested the Diverse Keyword Searchable Encryption (DKSE) method. The method obtains many practical applications for Multidimensional Numeric Vector Range Queries (MDDVRQ) and Textual Multi-Keyword Ranking Search (TMKRS).

H. Su et al. [11], The author proposed that the platform was tested on different Remote Patient Monitoring (RPM), providing AI-based anomaly detecting symptoms. This model achieves fast and accurate treatment for the patient. Ensures the information is confidential. R. Vargheese [12], In his article, the author describes Cisco's Cloud Web Security (CWS) method. In CWS, Cisco Global Threat Intelligence (CGTI), Advanced Threat Defense (ATPC), and Roamer

Protection (RUP) are used to protect users from anywhere, anytime in the distributed enterprise.

A. Jain et al. [13], The author propose securing and retrieving data access for Biometric Signature Authentication (BSA) scheme being presented using a Recurrent Neural Network (RNN). S. MR et al. [14], The author describes that the method does not need human help to produce the result. A powerful machine learning method extracts meaningful information for understanding big data.

M. Aruna et al. [15], The author propose the Cloud-based Intelligent Health Monitoring System (CIHMS) for disputes involving parties and manipulating sensitive records. Y. C. Yau et al. [16], The author describes Using the pattern-based data sensitivity framework (PBDSF) method. The machine learning algorithm can recognize specific patient data, data frequency, and different pattern codes identified under particular conditions to protect critical records.

Y. Yao et al. [17], the author propose A neural network (N.N.) computational approach that is applied to this model-based electronic health information system in the privacy-preserving Non-Collusive Dual Cloud (NCDC). B. Xi et al. [18], the author, concentrate on introducing hidden backdoor functionality for a collaborative training framework to any local hospital in joint with the global model.

Y. Chen et al. [19], Based on a multi-class support vector machine (MCSVM) scheme, the author proposed a privacy-preserving medical diagnosis scheme (PBMD). Two encryption systems use this method: Trapdoors Public Key Cryptosystems (DT-PKCs) and Boneh-Goh-Nissim (BGNs). H. D. Hoang et al. [20], the author proposes The HoloCare system provides permission and remote access to health record data from another EMR, giving evidence from a patient's PHR for authentication.

2.1 Problem Factors

- Essential privacy concerns, clinical decision samples, and biomedical data have been protected as risks.
- The amount, speed, variety, completeness, and value of patient data collected through telehealth applications make their big data. Dealing with a collection of heterogeneous data is one of the biggest challenges.
- Healthcare organizations face a series of data breaches targeting their most vulnerable medical records.
- It is very challenging to store extensive data in the dataset. It does not produce health care records.

3. Proposed Method

This section explains a machine learning-based Maximum Decision Support Regression (MDSR) based advanced secure Data Encrypt Transmission (ASDET) approach for efficient data communication in cloud storage. Our method improves

healthcare by protecting the privacy and confidentiality of sensitive data and preventing threats.

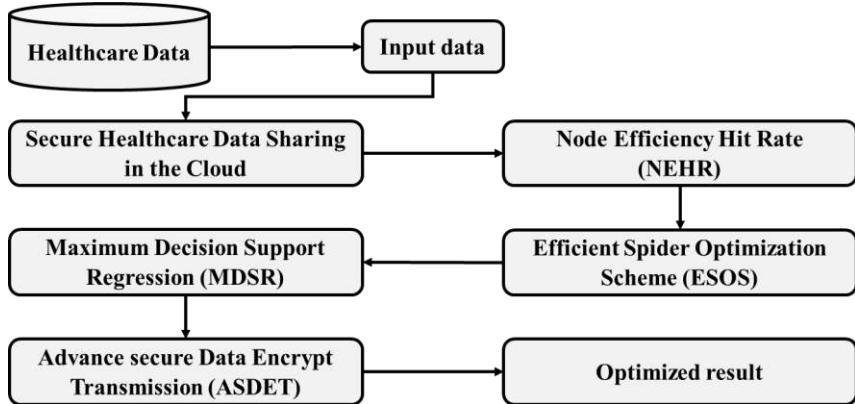


Fig. 1. Cloud-Based Secure Data Sharing for Health Care

Fig. 1. defines the proposed diagram for secure healthcare data sharing in the cloud environment. The proposed method initially analyses the node's trust, energy, delay, and mobility using the Node Efficiency Hit Rate (NEHR) method. Then identify the efficient route using an Efficient Spider Optimization Scheme (ESOS) for healthcare data sharing. After that, MDSR analyzes the malicious node for efficient data transmission in the cloud. The proposed Advanced secure Data Encrypt Transmission (ASDET) algorithm is used to encrypt the data.

3.1 Node Efficiency Hit Rate (NEHR)

In this phase, the proposed NEHR technique analyses the node trust, energy consumption, delay, and bandwidth. Every node calculates when sharing patient information in the cloud. The hit rate is the number of cache hits divided by the total number of memory requests for a given time interval. The NEHR algorithm first establishes various confidence factors based on the correlation between adjacent nodes, which observe each other. The node efficiency evaluation is based on the shared information between sources to destination without path traffic. And the energy consumption is found based on the total time and the total number of node energy. First, we analyze the trusted node calculation in equation 1.

$$T^n = \alpha_1 * C_{PF}(T) + \alpha_2 * D_P(T) \quad (1)$$

Let us assume α_1 and α_2 denotes weights allocated to control forward packet ratio (C_P) and forward data ratio (D_{PF}) at time T. If it is between [0.80 to 1], it is

considered a confidence node, and less than that is regarded as a less confidence node.

$$E^{Cons} = \frac{E^{Transmit} * E^{Rece}}{\beta} \quad (2)$$

Here the equation is used to calculate the node's energy consumption. Where $E^{Transmit}$ which refers to transmission and E^{Rece} denotes receiver energy and β refers to normalizing factor.

$$Tra^{Delay} = \sum \frac{Size(H^D)}{n} \quad (3)$$

The above equation is used to identify the health care data transmission delay Tra^{Delay} . Here H^D refers to health care data size, and n refers to number of nodes.

$$T^{mobility} = \sum Es_{mobility} + E^{Cons} \quad (4)$$

Here the equation is used to analyse the trusted mobility $T^{mobility}$ based on estimation of mobility $Es_{mobility}$ and Energy consumption E^{Cons} .

$$T^{reliable} = \sum D_f + E^{Cons} \quad (5)$$

Here the equation is used to find the trust node reliability $T^{reliable}$ based on data fusion D_f and Energy consumption E^{Cons} . This section calculates each node's reliability, mobility, latency, and energy consumption while sharing healthcare data in the cloud.

3.2 Efficient Spider Optimization Scheme

In this module, based on the spider optimization scheme, we find the node's energy consumption, distance, rate of sending packets, rate of received packets, and node availability, thereby designing the routing to share health data in the cloud. The proposed ESOS is an optimization technique inspired by the foraging behavior of spiders. Spider optimization is finding the next vibration based on the following maximum nearest weights of the features. And estimated the information in the feature selection part. The spider moves to a new node position and the resulting vibrations travel through the web. This vibration is generated by each spider and stored as node information, and other spiders on the web receive information based on the vibrations created by PPDT and FAHP spiders.

$$Dis_{node} = \sqrt{|l_1 - l_2|^2 + |O_1 - O_2|^2} \quad (6)$$

The above equation is used to find the distance between two nodes using Euclidean distance. The Initial location of two nodes angel (l_1, l_2) and (O_1, O_2) respectively.

$$E^{Resi} = \frac{i^{Energy} - E^{Cons}}{i^{Energy}} \quad (7)$$

The above equation finds each node's residential energy (E^{Resi}). Let us assume I^{Energy} it represents initial energy and E^{Cons} energy consumption.

$$S^W = \frac{Fitness_i - W_t}{b_t - W_t} \quad (8)$$

The above equation is used to find the spider weightage W_t , $Fitness_i$ which denotes fitness of i^{th} spider position and W_t , b_t denotes worst and best fitness respectively.

$$S^{vibration} = S^W * e^{Dis_{node}(y,z)} \quad (9)$$

The above equation finds spider vibration ($S^{vibration}$) to choose best node for healthcare data sharing in the cloud environment. Here we assume x and z denote two neighbour nodes.

$$RP_{rate} = \sum \frac{N(R_p)_{y,z}(T) - (R_p)_{y,z}(T-1)}{N(R_p)_{y,z}(T) + (R_p)_{y,z}(T-1)} \quad (10)$$

The above equation analyses the packet received rate RP_{rate} at time T. Where N refers to number nodes. R_p Denotes healthcare data received nodes.

$$SP_{rate} = \frac{SP_N(y,z)(T)}{SP_N(y,z)(T) + SP_{y,z}(T)} \quad (11)$$

The above equation is used for healthcare data sending rate (SP_{rate}) based on needing several sent packets SP_N and a repeated number of transmitted packets SP_r .

$$N^{avail} = \frac{Ack_{yz}(T)}{Ack_{yz}(T) + NAck_{yz}(T)} \quad (12)$$

The above equation is used to analyse node availability (N^{avail}) acknowledgement at time T. Here assume Ack is the responding node and $NAck$ is the non-respond node at time T. This module efficiently creates the route for healthcare data sharing in a cloud environment using ESOS. This method proficiently analysis the route based on spider vibration, and this vibration has an essential feature of the node.

3.3 Maximum Decision Support Regression

The proposed Maximum Decision Support Regression (MDSR) detects the essential features of the node in this section. It considers the packet delivery rate, efficiency and fitness function as important aspects of the node. This method takes only the most supported results. Then the proposed method identifies whether the node is malicious or not.

Algorithm for Maximum Decision Support Regression (MDSR)

Begin function

 Import the node population
 For each node $N_{y,z} = 1$ do

```

Calculate the packet delivery ratio ( $P_{ratio}$ )

$$P_{ratio} = \sum \frac{Rece_d}{sent_d}$$

Calculate throughput performance ( $Through^P$ )

$$Through^P = \sum \frac{S_T}{T}$$

If identify the intruder node  $Intruder^{node}$ 

$$Intruder^{node} = 1 - \left( \frac{Rece^{node} - Com^{node}}{R_n} \right)$$

End if
Compute node fitness function  $Node^{Fit}$ 

$$Node^{Fit} = \alpha_1 E^{node} + \alpha_2 \frac{1}{TraDelay} + \alpha_3 \frac{1}{Disnode}$$

End for
End function

```

The proposed algorithm steps identify the node features based on throughput $Through^P$, packet delivery ratio (P_{ratio}) and check malicious node during healthcare data sharing in the cloud. Here we assume $Rece^{node}$ denotes the received node, Com^{node} compromised node and R_n is the round number. S_T Denotes packet successfully sent at time T. Then α_1 , α_2 , α_3 weight coefficients and E^{node} is the energy node.

3.4 Advance secure Data Encrypt Transmission (ASDET)

The proposed Advanced secure Data Encrypt Transmission (ASDET) method in this module facilitates the secure transport of health data. First, it reads the health data as input, generates a key for it, converts it into a binary value corresponding to the characters in the data, and then converts it into ASCII values. The proposed method transforms that data into an unreadable format. The proposed Advanced Secure Data Encrypt Transmission (ASDET) technique is used to convert plain health care data into cipher text C^{Text} proficiently.

Algorithm Steps for Advance Secure Data Encrypt Transmission (ASDET)

```

Begin function
    Read input healthcare data ( $H_d$ )
    For each node  $N_{(y,z)}=1$  do
        Randomly Generate key  $G_k = (p_k, s_k)$ 
        Separate the plaintext into two and calculate the Binary value
    Compute equivalent values
         $C^{Text} = H_d + G_k \text{ mod } 256$ 
        Sort the  $C^{Text}$ 
    End for
Stop function

```

Here assume that p_k, s_k are private key and public key respectively. This proposed technique efficiently secure the health care data in the cloud environment.

4. Result and discussion

This section describes the simulation results of the proposed and existing algorithms by analyzing and evaluating them. The proposed algorithm Maximum Decision Support Regression based Advance secure Data Encrypt Transmission (MDSR-ASDET) and the existing algorithms such as Privacy Preserving Decision Tree (PPDT) and Fuzzy Analytical Hierarchy Process (FAHP) simulation parameters have been evaluated.

Table 1: Simulation Parameters

Parameters	Values
Tool name	Visual Studio 2012
Front End	Asp.Net
Number of data	40
Traffic	Constant Bit Rate (CBR)
Cloud Type	AWS
File size	25GB, 50GB and 75 GB

Table 1 defines the simulation parameters of the proposed implementation in Visual Studio 2012. The proposed algorithm parameters are throughput performance, latency performance, security and cloud storage. Based on the health data attribute, several data are counted, so the count takes from 10-40.

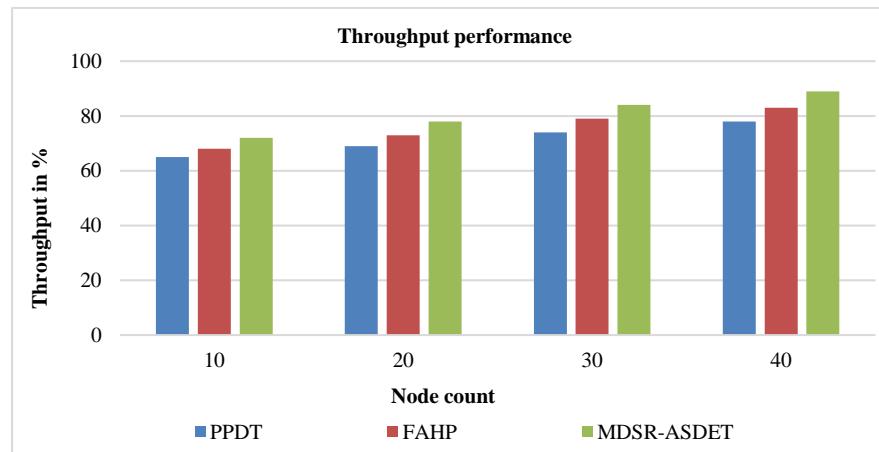


Fig. 2. Analysis of Throughput Performance

Fig. 2. illustrates the performance of the OBS network when it comes to throughput. According to our analysis, the proposed MDSR-ASDET algorithm achieves 89% throughput performance. In addition, the current algorithms perform better than the existing algorithms in terms of

$$\text{Throughput Performance} = (\text{number of data requests}) / (\text{total time})$$

The throughput performance evaluates the packet delivery ratio based on the number of packets sent to the destination divided by the overall packet count. The proposed method is 89% the same as the throughput and delivery ratio in throughput performance.

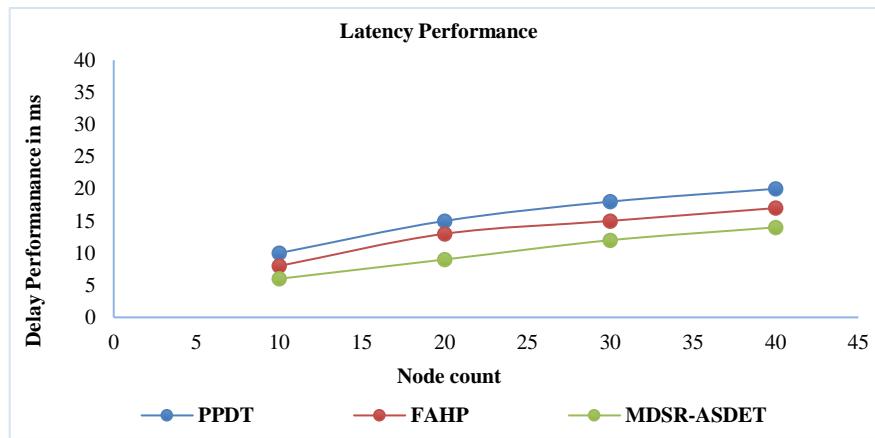


Fig. 3. Analysis Latency Performance

Fig. 3. defines the latency performance for healthcare data sharing in the cloud via nodes. The proposed MDSR-ASDET algorithm result is 14ms; Additionally, the existing algorithm results are 20ms for the PPDT and 17ms for FAHP, respectively.

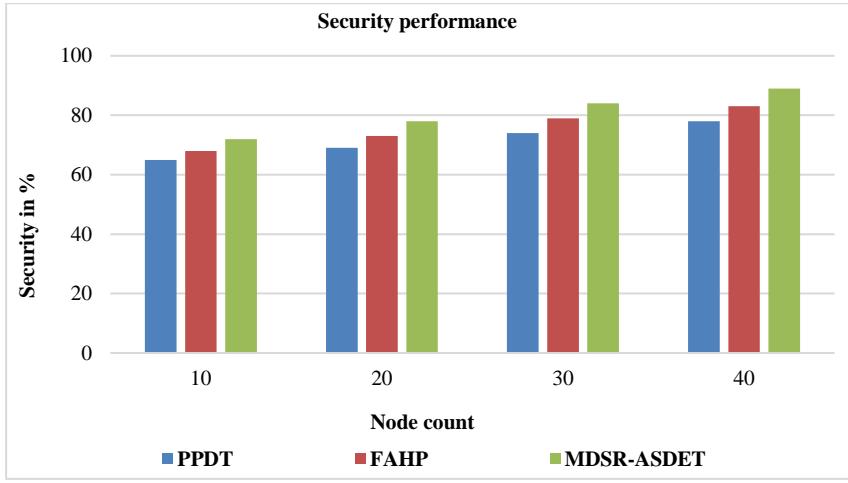


Fig. 4. Comparison of Security Performance

Fig. 4. defines the graph's comparison of healthcare data sharing security performance results. In the chart analysis, the proposed method produces better security performance than previous methods. The proposed MDSR-ASDET technique has a security result is 92%.

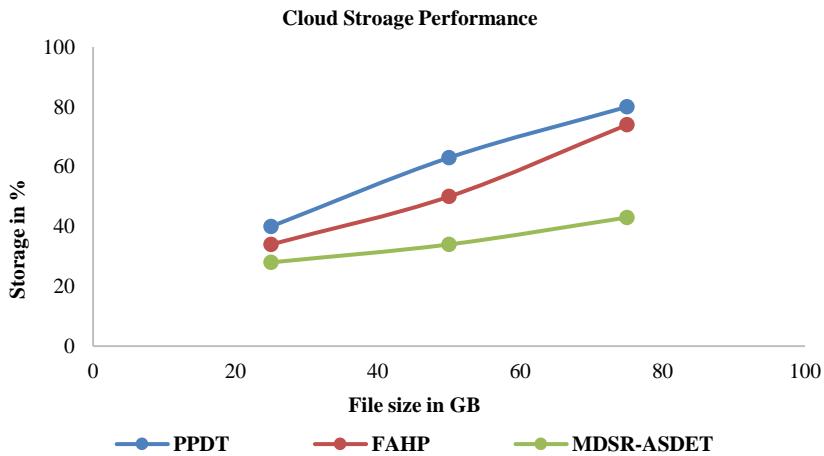


Fig. 5. Result of Cloud Storage

Fig. 5. explores the healthcare cloud storage data performance of the proposed and existing performance shown in the graph. The proposed method cloud storage result has 43% for 75GB files. Similarly, the existing PPDT result was 80%, and the FAHP result was 74% for 75GB files.

5. Conclusion

To conclude, this paper introduced a machine learning-based Maximum Decision Support Regression (MDSR) based advanced secure Data Encrypt Transmission (ASDET) approach for efficient data communication in cloud storage. Initially, the proposed method analysis the node's trust, energy, delay, and mobility using Node Efficiency Hit Rate (NEHR) method. Then identify the efficient route using an Efficient Spider Optimization Scheme (ESOS) for healthcare data sharing. After that, MDSR analyzes the malicious node for efficient data transmission in the cloud. The proposed Advance secure Data Encrypt Transmission (ASDET) algorithm is used to encrypt the data. Simulation results produced by the proposed method are superior to previous methods.

Reference

- [1] Jindal, A. Dua, N. Kumar, A. K. Das, A. V. Vasilakos and J. J. P. C. Rodrigues.: Providing Healthcare-as-a-Service Using Fuzzy Rule Based Big Data Analytics in Cloud Computing. IEEE Journal of Biomedical and Health Informatics. vol. 22, pp. 1605-1618. (2018)
- [2] X. Li, S. Liu, R. Lu, M. K. Khan, K. Gu and X. Zhang.: An Efficient Privacy-Preserving Public Auditing Protocol for Cloud-Based Medical Storage System. IEEE Journal of Biomedical and Health Informatics. vol. 26, pp. 2020-2031. (2022)
- [3] Agrawal et al.: Evaluating the Security Impact of Healthcare Web Applications Through Fuzzy Based Hybrid Approach of Multi-Criteria Decision-Making Analysis. IEEE Access. vol. 8, pp. 135770-135783. (2020)
- [4] L. Wang, Y. Ali, S. Nazir and M. Niazi.: ISA Evaluation Framework for Security of Internet of Health Things System Using AHP-TOPSIS Methods. IEEE Access, vol. 8, pp. 152316-152332. (2020)
- [5] M. Zarour et al.: Evaluating the Impact of Blockchain Models for Secure and Trust-worthy Electronic Healthcare Records. IEEE Access, vol. 8, pp. 157959-157973. (2020)
- [6] H. Abrar et al.: Risk Analysis of Cloud Sourcing in Healthcare and Public Health Industry. IEEE Access, vol. 6, pp. 19140-19150. (2018)
- [7] S. More et al.: Security Assured CNN-Based Model for Reconstruction of Medical Images on the Internet of Healthcare Things. IEEE Access, vol. 8, pp. 126333-126346. (2020)
- [8] Y. K. Saheed and M. O. Arowolo.: Efficient Cyber Attack Detection on the Internet of Medical Things-Smart Environment Based on Deep Recurrent Neural Network and Machine Learning Algorithms. IEEE Access, vol. 9, pp. 161546-161554. (2021)
- [9] W. Niu, J. Huang, Z. Xing and J. Chen.: Knowledge Spillovers of Medical Big Data Under Hierarchical Medical System and Patients' Medical Treatment Decisions. IEEE Access, vol. 7, pp. 55770-55779. (2019)
- [10] X. Wang, J. Ma, Y. Miao, X. Liu and R. Yang.: Privacy-Preserving Diverse Keyword Search and Online Pre-Diagnosis in Cloud Computing. IEEE Transactions on Services Computing, vol. 15, pp. 710-723. (2022)
- [11] H. Su et al.: Cloud Computing Management Architecture for Digital Health Remote Patient Monitoring. In: 2021 IEEE International Conference on Smart Computing (SMARTCOMP), pp. 209-214, (2021)

- [12] R. Vargheese.: Dynamic Protection for Critical Health Care Systems Using Cisco CWS: Unleashing the Power of Big Data Analytics. In: 2014 Fifth International Conference on Computing for Geospatial Research and Application, pp. 77-81, (2014)
- [13] Jain and K. Tripathi.: Biometric Signature Authentication Scheme with RNN (BIOSIG_RNN) Machine Learning Approach. In: 2018 3rd International Conference on Contemporary Computing and Informatics (IC3I), pp. 298-305, (2018)
- [14] S. M.R, M. T.N, A. R.A and R. S. Hegadi.: Enhanced Efficient and Security in Big Data Using TDES And Machine Learning Technique. In: 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), pp. 1-6, (2022)
- [15] M. Aruna, V. Arulkumar, M. Deepa and G. C. P. Latha.: Medical Healthcare System with Hybrid Block based Predictive models for Quality preserving in Medical Images using Machine Learning Techniques. In: 2022 International Conference on Advanced Computing Technologies and Applications (ICACTA), pp. 1-10. (2022)
- [16] Y. C. Yau, P. Khethavath and J. A. Figueiroa.: Secure Pattern-Based Data Sensitivity Framework for Big Data in Healthcare. In: 2019 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD), pp. 65-70, (2019)
- [17] Y. Yao, Z. Zhao, X. Chang, J. Mišić, V. B. Mišić and J. Wang.: A Novel Privacy-Preserving Neural Network Computing Approach for E-Health Information System. In: ICC 2021 - IEEE International Conference on Communications. pp. 1-6. (2021)
- [18] B. Xi, S. Li, J. Li, H. Liu, H. Liu and H. Zhu.: BatFL: Backdoor Detection on Federated Learning in e-Health. In: 2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS), pp. 1-10. (2021)
- [19] Y. Chen, Q. Mao, B. Wang, P. Duan, B. Zhang and Z. Hong.: Privacy-Preserving Multi-Class Support Vector Machine Model on Medical Diagnosis. IEEE Journal of Biomedical and Health Informatics, vol. 26, pp. 3342-3353, (2022)
- [20] X. Zheng, R. R. Mukkamala, R. Vatrapu and J. Ordieres-Mere.: Blockchain-based Personal Health Data Sharing System Using Cloud Storage. In: 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 1-6. (2018)

Routing Integrity Mechanism to Prevent Wormhole Attacks In Vehicular Adhoc Networks

*Prathap Kumar R¹, U Srilakshmi², Ganesh Reddy K³

¹VFSTR Deemed to be University

¹rpk_cse@vigann.ac.in

²VFSTR Deemed to be University

²druppalapati2019@gmail.com

³VIT-AP Universtiy

³ganesh.reddy@vitap.ac.in

Abstract. The field is experiencing an increase in research finding due to the significance of VANETs in the contemporary world. Routing is essential for vehicle communication in a VANET, but choosing the best route can be challenging given how quickly cars move. On these routing paths, it is still challenging to prevent attacks like wormhole, grayhole, and sinkhole attacks. Researchers have created defensive mechanisms, but because routing packets frequently lack message integrity, they are insufficient. In this paper, we have proposed a delay-sensitive routing parameters with integrity mechanism to detect and prevent the wormhole attack. In this study, we found vulnerable routing parameters for wormhole attack detection and secured them with hash functions. Our tests demonstrate that vehicle communication protect against wormhole attacks. This implies that throughput is higher and more packets are delivered even in hostile environments.

Keywords: Vehicular Adhoc Networks, Vehicle to Vehicle, Vehicle to Infrastructure, Wormhole attack, source, destination, throughput, Round Trip Time

1 Introduction

According to the WHO, vehicle accidents are among the top five causes of mortality and property damage (World Health Organization). According to the World Health Organization, 1.35 million people are killed in road accidents each year, with an additional 20 to 50 million wounded in varying degrees of seriousness. The world is split into three parts: The three economic kinds are high-income nations, middle-income countries, and low-income countries. Accidents are highly prevalent in the low and middle classes (approx 93 percent). Traffic accidents have taken thousands of lives and billions of dollars throughout the world, forcing the improvement of new technology and techniques to help in their prevention.

Several attempts in the literature have been made to find technology that can prevent accidents, and VANET solutions have been advocated. Due to its security, VANET has risen in popularity. ITS, MANET, and IoT are all part of the system. While several wireless network research topics have been explored, VANET has emerged as the key field of study.

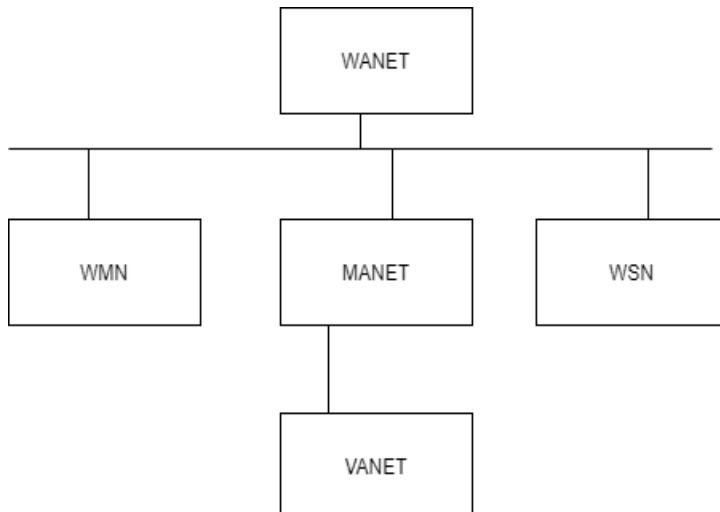


Fig. 1. TYPES OF WANET

The network is made of many vehicles and other devices that may communicate with one another and exchange information. VANET has emerged as a viable subclass or variation with the advent of the MANET class. A VANET secures the safety of both the driver and the passengers, as well as a pleasant ride for both. VANETs link with roadside devices and other cars by using real automobiles as nodes. The communication protocols V2V and V2I are utilized in VANET. The network is made up of several cars and other equipment that may communicate and share data with one another. utilizing MANET concepts, but with the automobiles doing additional functions such as nodes. The data of each vehicle will be kept and shared by all vehicles. After all network nodes have exchanged information, the information is collected and combined before being sent to all connected devices. You can join and exit the VARENT network whenever you choose. While VANET has mostly been utilized for safety purposes, it is sometimes used for non-safety purposes. Riders can be advised and warned of a range of concerns, such as an accident, detour, or avoidable collision. There is a shortage of data on traffic, parking, gas stations, and hotels. The whole network architecture is represented. Proceedings in Information and Communication Technology (PICT).

VANET Architecture

The VANET design aims to deliver pertinent information to the network's nodes and routers. Every piece of data needed to create and sustain a vehicle is kept in each one, updated anytime something changes, and distributed throughout the network. Each car acts as a sender, receiver, or router in this scenario. V2V, V2I, V2V, and V2I constitute one of the four categories for VANET architecture [7]. Information can be sent via a variety of methods, which the transmitter and receiver of the communication exchange back and forth [8]. You must incorporate the following in order to create a VANET network: "Application Unit," "On Board Unit," and "Remote Support Unit" (Road side Unit).

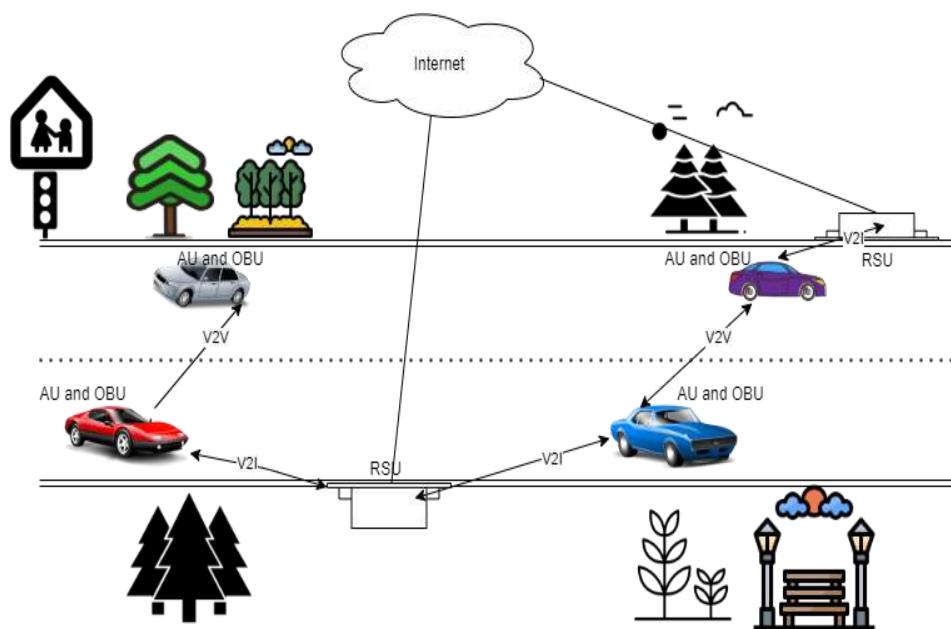


Fig. 2. VANET Architecture

External wormhole attack in VANET

An external wormhole attack attempts to induce network nodes to route their traffic through a long, malicious tunnel that is broadcast across the network with low latencies and a low hop count. At least two attacker nodes with an illegitimate long communication link are required to carry out this attack. The attacker node at one end of the malicious tunnel receives data from its neighboring nodes and sends it to the attacker at the other end of the tunnel. Rather than forwarding these packets, the attacker drops/injects malicious packets into the data traffic.

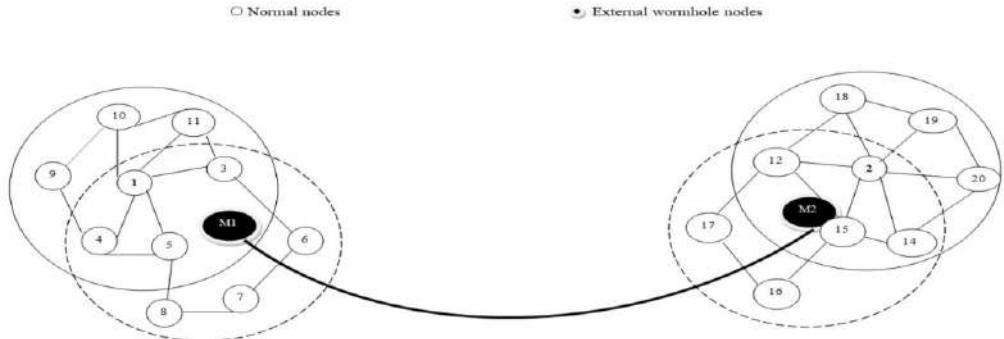


Fig. 3. External malicious vehicles formed the wormhole.

Internal Wormhole attack in VANET

The goal of an internal wormhole attack is to attract network nodes to forward their traffic through a long, malicious tunnel that is broadcast over the network with low latencies and less hop count. In order to carry out this attack, at least two attacker nodes are required. One attacker node encapsulates the request packet, and the other attacker node decapsulates it. The actual hop count and latency of the packet are not updated by the other intermediate node(s) in the routing path throughout the encapsulation and decapsulation operations. Here, internal wormhole attackers exploit the attack on the integrity of the route request and reply messages.

Compared to external wormhole attacks, internal wormhole attacks are easy to create without any special resources like long-distance communication links. However, internal wormhole attacks are difficult to detect because of the nodes' mobility and the attacker's ability to encapsulate or encapsulate the route messages. In this paper, we address the both internal and external wormhole attacks.

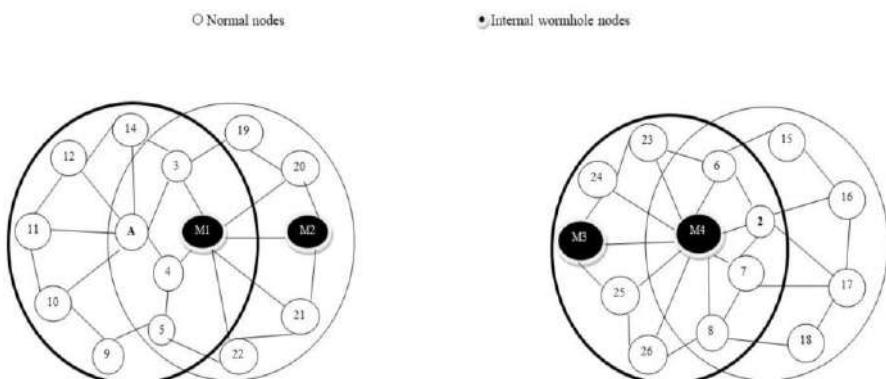


Fig. 4: Internal malicious vehicles formed the wormhole

2 Literature

We have studied the VANET security mechanisms under 3 categories in this section . First one is detection mechanisms, second one is prevention mechanisms and third one is both prevention and detection mechanism.

AlFarraj et al. [9] proposed the dependable neighboring group nodes on their work with the activation function. The plan's goal was to strengthen the security measures that were already in place. The authors used energy consumption to calculate the significance of the weight of confidence. We were able to estimate the node using an additive measure without jeopardising the dependability of the node's neighbouring nodes.

Ahmed et al. [10] proposed TESRP, which stands for Trust and Energy-aware Secure Routing Protocol. [Citation required] In this method, the authors used a distributed trust model to identify bad actors. When it comes to routing, a strategy that incorporates a number of different factors has been shown to be the most successful. The authors were able to significantly improve the network's energy efficiency. Mehetre and colleagues [11] proposed a method for secure routing that is based on mutual trust between network nodes. The authors recommend using a two level security and two level trust planning system to select the node, and to encrypt the data packet. For the reliable communication authors used Cuckoo Search Algorithm.

Innovative technological solutions put forward by Tyagi et al. [12] to evaluate the effectiveness of VANET safety measures The AODV protocol has been altered in response to the findings of the investigation into the blackhole attack. The RREP packet has been altered to improve its ability to detect blackhole attacks, and all of the data pertaining to route replies has been moved to a lookup table. A cross-layer detection strategy that is based on the Link State Routing (LSR) protocol was developed by Baiad et al. [13] with the intention of improving the quality of service (QoS). A monitoring node is used in order to determine whether or not a data packet is valid. As a consequence, it was discovered that the protocol outperforms previous ways of detecting black holes while also lowering the amount of false positives.

According to Parmar Amish et alidea, .'s a wormhole attack may be thwarted provided each node stores crucial information about its neighbours in its routing table. If a route cannot be found, the node will search for the information in the route table it keeps, send out a response packet, and wait for a response. A node will resend a request packet along the same path that it originally sent it along after receiving it at that node. The sender will conclude that there is a substantial amount of road available if sender receives more than one reply packet. When the estimated round-trip time (RTT) is below the threshold, the sending node will launch a wormhole attack by dropping those routes [14].

The presented scheme uses AODV. Simulations with 100 nodes showed high detection accuracy without storage needs. The study cited in Reference (Ref.) [15] presented its findings as an energy-saving wormhole defense scheme (EPSMAW). EPSMAW reduces end-to-end delay, energy use, and traffic

overhead. The proposed solution uses AODV routing and is based on neighbor and connectivity information. Simulations with 150 nodes showed high throughput and low false positives.

[16] presented a software-defined network-based method for wormhole detection. It uses neighbour similarities. Replicas Python-written method tested on 100,000-node network After computing the NSI and ACI, the K-means clustering method was used. SWAN can detect wormholes with low communication overhead, FPR, and FNR.

[17] presented a wormhole detection scheme for 3D networks. This scheme was based solely on node connectivity. The MAXIS algorithm is greedy. The suggested method is simple. Detection rate was calculated using different node densities. The findings showed the method's 90% accuracy. Greedy algorithms can't find the best solution. According to Reference [16], NIAPC was proposed as a solution that offers high accuracy, PDR, and throughput.

A cluster-based model has been proposed as a MANET defence mechanism against wormhole attacks [18]. In particular, a location-based geo-casting and forwarding (LGF) protocol and a k++means clustering algorithm were used to ensure the highest level of confidentiality and to find the most time and resource-efficient route. Load balancing, delivery ratio, and delay from beginning to end were used to evaluate performance. Furthermore, the AODV protocol's dynamic routing resulted in an unprotected send and receive. Wormholes posed a significant challenge due to the possibility of node intrusion during the dynamic routing request process. An energy model [19] was implemented to help in the detection of wormhole attacks. We determined how much power could be transmitted at once and how much energy each node had during this stage of the process.

In [20], deep learning is proposed to be used in the development of an intrusion detection system (IDS) for on-board units. A PCAP file is created to store data before it is processed. Deep learning techniques such as LSTM and CNN are used here. After the data has been extracted using a CNN, machine learning is performed using a three-layered LSTM while taking the temporal context of the data into account. The LSTM uses the results generated by the CNN as its data source. DeepVCM is composed of two layers of convolutional neural networks (CNN), two layers of maxpooling, two layers of local response normalisation, and three layers of long short-term memory (LSTM). Max-pooling receives the results generated by the CNN layer as an input. The most recent version of DeepVCM can be downloaded and installed automatically. Despite having to work within a limited set of constraints, the proposed model produced impressive results.

As part of their intrusion detection system (IDS) proposal, the authors of [21] employ a dynamic neuro fuzzy system to search for evidence of routing attacks. The scalability of individual nodes as well as the overall network reach are being investigated and analysed. The simulations are run using the software MATLAB. In terms of throughput, average download relay, end-to-end delay, and packet loss rate, the proposed method outperforms the existing methods.

In [22] , A novel clustering-based optimization technique is proposed. It has been

observed that it will improve the effectiveness of V2V communication even further. In this paper, the vehicle nodes are clustered using the K-Medoid clustering model and then used to improve energy efficiency. To establish an energy efficient communication methodology, a metaheuristic algorithm is used. Based on the simulation analysis, it is clear that this methodology requires less execution time and improves the energy efficiency of the nodes.

Based on our study, we observed that existing detection mechanisms for wormhole attacks in VANET have a high false positive and false negative rate. Most existing security mechanisms do not consider vehicle location and message integrity when identifying malicious functionalities, resulting in inadequate detection and prevention of wormhole attacks on VANET.

3. Delay sensitive Message integrity Mechanism to Isolate the Wormhole attacks

Detect and prevent wormhole attack, we have added the timestamp-based message integrity in the route request message. Initially, nodes need to identify the average communication link delay(avg_del) in VANET,

$$\text{Avg}_{\text{delay}} = \frac{\sum_{i=1}^{n-1}(p_i+t_i)}{n-1} + \frac{\sum_{i=1}^n(\text{pro}_i+q_i)}{n} \quad \dots\dots(1)$$

Where n is the number of nodes, p_i propagation delay, t_i transmission delay, Pro_i processing delay and q_i queuing delay.

We have considered the source IP address, destination IP address and timestamp of route request/ reply message to create message digest. We use SHA-256 algorithm to create message digest by considering these three parameters

$$H(M) = \{\text{Source IP}, \text{Destination IP}, \text{Timestamp}\} \dots\dots(2)$$

Route Request (RREQ)/ Route Reply (RREP) packet: General fields in the route request and reply packets are Source IP, destination IP, Sequence number, hop_count, timestamp, and Time To Live (TTL). In addition to that we have added the message digest to provide message integrity.

Source IP	Destination IP	Sequence number	Hop_count	Timestamp	TTL	H(M)
-----------	----------------	-----------------	-----------	-----------	-----	------

RREQ and RREP packets use H(M) for verification

When the source vehicle (S) needs to form a route to the destination vehicle (D) in our proposed algorithm, S must first find the path to D. S broadcasts the route request packet in order to find the route to the destination node. Request packets contain the request packet's timestamp and message digest, which cannot be changed by any other network vehicle. Before broadcasting into the network, all intermediate nodes update the hop count. In our proposed work, intermediate

vehicles do not need to perform request message verification, which reduces the computational overhead at intermediate vehicles. When the D receives a request packet, it verifies the integrity of the request packet (RREQ) using a hash function and the minimum hop count in relation to time. If both conditions are met, D generates a reply message and forwards it to S via the reverse path. If one or both of the conditions are not met, D drops the RREQ packet due to a wormhole attacker in the path. Similarly, upon receiving the reply packet, S verifies the replay packet integrity using a hash function and the minimum hop count in relation to time. If both conditions are met, S chooses the path to forward the data packets, as explained in algorithm 1.

Algorithm: Delay sensitive Message Integrity Mechanism to Isolate the Wormhole attacks

Inputs: Route Request and replay messages, number of vehicles(n), malicious tunnels, and legitimate links

Output: detect and isolate malicious tunnels

1. Source vehicle broadcast RREQ packet to find the route to destination
2. If(source IP= Receiver IP)
 3. Drop the RREP packet
 4. else if (Source IP ≠ Receiver IP && Destination IP ≠ Receiver IP)
 5. Hopcount ← Hopcount+1
 6. broadcast the RREQ packet
 7. else if (Destination IP = Receiver IP)
 8. if($\text{Avg}_{\text{delay}} > (\text{rtimestamp}-\text{timestamp})/n$)
 9. Current path between source and destination contains wormhole attack
 10. Drop the RREQ packet
 11. else
 12. Destination vehicle sends RREP to source vehicle in the reverse path
 13. /*Reply packet*/
 14. if (Source IP ≠ Receiver IP)
 15. Forward the RREP packet to next router
 16. else if (Source IP = Receiver IP)
 17. if($\text{Avg}_{\text{delay}} > (\text{rtimestamp}-\text{timestamp})/n$)
 18. Current path between source and destination contains wormhole attack
 19. Drop the RREP packet
 20. else
 21. NO internal worm hole existing between source and destination path

4. RESULT ANALYSIS

We have created a VANET scenario using SUMO tool then the code converts in to tcl file which is directly executed in the network simulator (ns2). NS2 environment was used to run VANET scenario and test our proposed notion in a hostile environment. In a network, we believe there are fifty valid nodes. UDP

traffic is used by network nodes to communicate. Eight rushing attackers will attack the UDP traffic. According to the network dimensions, all vehicles are configured with 802.11p MAC protocol. Table 1 shows a number of network setups. We have considered the ad-hoc on demand distance vector routing protocol for path selection, and to avoid the congestion, we used Drop-tail queue. The vehicles move in the 5000m X 5000m coverage area. The vehicles use a random-way point mobility model, and the total simulation time is 100 sec for attack, non-attack, and proposed systems.

Table 1. Simulation Parameters

PARAMETER	VALUE
Non- Malicious nodes	50
Routing Protocol	AODV
MAC	802.11p
Queue Type	Drop-tail
Packet Size	512 bytes
Transport protocol	UDP
Network area	5000m X 5000 m
Mobility	Random-Way point
Number of wormhole attackers	8
Simulation Time	100 sec

We have implemented the wormhole attacks in which the one-end of the wormhole tunnel attacker receives the packets from the source vehicles and forward these packets to the other end wormhole attacker. Upon receiving the packets, the attacker performs the

- Dropping all packets
- Dropping selective packets
- Reordering packets
- Injecting malicious packets
- Increase the queuing delay of each packet

Throughput: The throughput is defined as the number of data packets successfully transmitted per Unit-time. We have compared our proposed **delay sensitive routing parameter with integrity** throughput with the Round Trip

Time (RTT) mechanism[7] and attacks scenarios[12] which is shown in figure x. Based on our results, our proposed **mechanism** has better throughput then attack scenario and some our proposed approach has shown better performance than non-attack scenario due to selecting best paths.

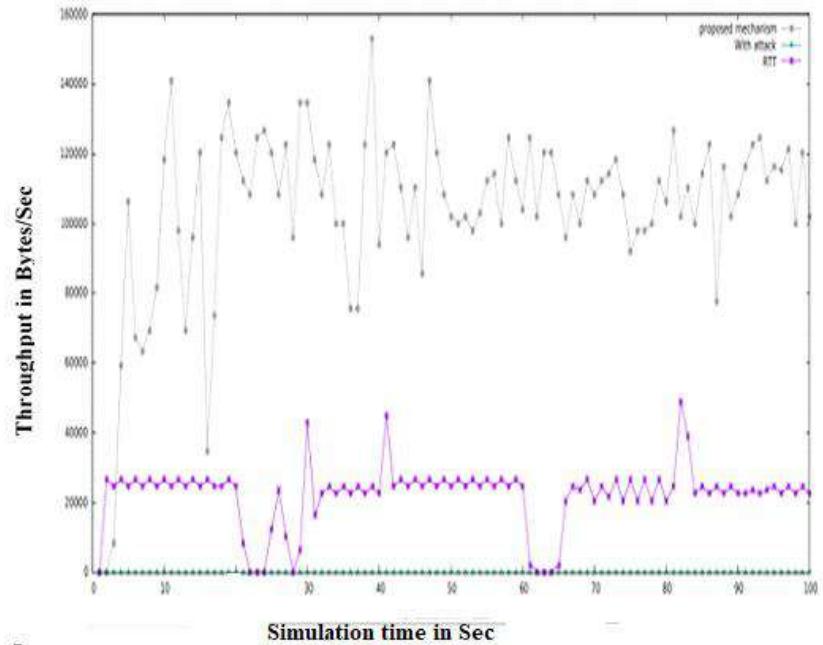


Fig. 5: Throughput comparision analysis

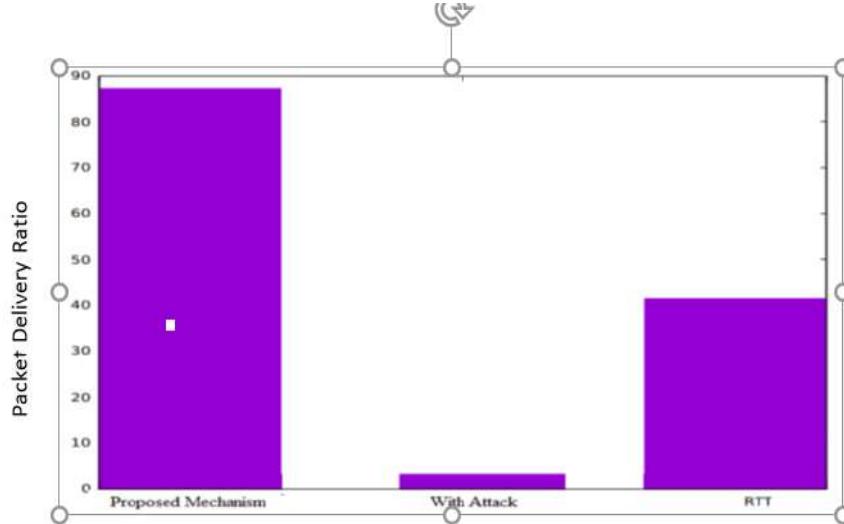


Fig. 6: PDR Comparision analysis

Packet Delivery Ratio (PDR): We also checked the network performance in terms of PDR. We observed that the PDR value is 3% in the attack scenario and 41% for RTT approaches. The PDR value in our proposed approach is 87%. This shows our proposed approach out performs the attack and RTT approaches. This is because of prevention of long distance paths and isolating the more collision prone links in the communication, and these analysis results are shown in above figure.

5. Conclusions

In this paper, we have identified the shortcomings of existing wormhole detection mechanisms and propose delay-sensitive routing parameters with an integrity mechanism to prevent internal wormhole attacks in VANET. Internal wormhole attacks are more efficiently detected and isolated in our proposed mechanism because both route request and reply messages are verified with minimum delay values and any modification of routing parameters at each node in the active path. In our experimental analysis, we compared our proposed mechanism to the existing solutions, and we discovered that our proposed mechanism isolates the wormhole attacks immediately after they happen. As a result, our proposed approach provides five times better throughput than existing mechanisms and an 87% packet delivery ratio in an attack environment. In our future work, we need to consider cross-layer metrics like vehicle communication range, channel error rate from the physical layer, and queue size from the MAC layer to detect wormhole attacks.

References

1. Vamshi Krishna K, Ganesh Reddy:A Delay Sensitive Multi-Path Selection to Prevent the Rushing Attack in VANET, 5th International Conference on Information Systems and Computer Networks (ISCON), IEEE Conference ,India. (2021)
2. K. Ganesh Reddy, P. Santhi Thilagam: Trust-based hybrid ids for rushing attacks in wireless mesh Networks, Recent Advances in Computer Based Systems, Processes and Applications: Proceedings of Recent Advances in Computer based Systems, Processes and Applications (2019)
3. Parul Tyagi, Deepak Dembla, "Performance analysis and implementation of proposed mechanism for detection and prevention of security attacks in routing protocols of vehicular ad- hoc network (VANET)", Egyptian Informatics Journal, Volume 18, Issue 2, Pages 133-139, ISSN 1110-8665 (2017).
4. Prathap kumar Ravula, M. Shanmugam: A Detailed Case Study on VANET Security Requirements, Attacks and Challenges, Advances in Modelling and Analysis B, IIETA, Scopus,Unpaid, Vol 62, No:2-4, pp-48-52, ISSN NO:1240-4543 (2020)
5. R. Prathap Kumar, D. Murli Krishna Reddy, G.Parimala, S. Deva Kumar, " IMPACTS OF WORMHOLE AND BLACK HOLE ATTACKS IN VANETS", Design Engineering , ISSN: 0011-9342, issue: 7,pp. 2128- 2146 (2021).
6. Parimala Garnepudi, Suneetha Vesalapu, Vidyullatha P, A V R Mayuri, R Prathap Kumar, "An Efficient Algorithm for Enhancing QOS in VANET", Turkish Online Journal of Qualitative Inquiry (TOJQI), Volume 12, Issue 3, 4420 - 4430, ISSN: 1309-6591 (2021).
7. Taleb :VANET Routing Protocols and Architectures: An Overview", Journal of Computer Science, 14(3), 423-434 (2018).
8. Tomar, Ravi & Prateek, Manish & Sastry, Hanumat: Vehicular Adhoc Network (VANET)- An introduction, 9, 8883-8888 (2016)
9. AlFarraj, Osama, Ahmad AlZubi, and Amr Tolba: Trust-based neighbor selection using activation function for secure routing in wireless sensor networks, Journal of Ambient Intelligence and Humanized Computing 1-11 (2018.)
10. Ahmed, A., Bakar, K., Channa, M., & Khan: A secure routing protocol with trust and energy awareness for wireless sensor network. Mobile Networks and Applications, 21(2), 272–285 (2016).
11. Mehetre, Roslin, & Wagh: Detection and prevention of black hole and selective forwarding attack in clustered WSN with Active Trust. Cluster Computing, 22, 1313, (2018).
12. Tyagi, P& Dembla: Performance analysis and implementation of proposed mechanism for detection and prevention of security attacks in routing protocols of vehicular adhoc network, Egyptian Informatics Journal, 18(2), 133–139 (2017).
13. Baiad, R., Alhussein, Otrok & Muhamadat: Novel cross layer detection schemes to detect black hole attack against qos-olsr protocol in vanet, Vehicular Communications, 5, 9–17 (2016).
14. Amish, P., & Vaghela: Detection and prevention of wormhole attack in wireless sensor network using AOMDV protocol, 7th international conference on communication, computing and virtualization (Vol. 2016, pp. 700–707) (2016).
15. Shahid, H., Ashraf, H., Ullah, A., Band, S. S., & Elnaffar : Wormhole attack mitigation strategies and their impact on wireless sensor network performance:A literature survey , International Journal of Communication Systems, e5311 (2022).
16. Hanif, Maria, et al. :AI-Based Wormhole Attack Detection Techniques in Wireless Sensor Networks, Electronics 11.15 : 2324(2022).
17. Bai, Sen, et al :Detecting wormhole attacks in 3D wireless ad hoc networks via 3D forbidden substructures, Computer Networks 150: 190-200 (2019).

18. Spurthy, K., & Shankar: An efficient cluster-based approach to thwart wormhole attack in adhoc networks, International Journal of Advanced Computer Science and Applications, 11(9), 312–316 (2020).
19. Gayathri, Seetharaman, Subramanian, Premkumar, Viswanathan & Chandru: Wormhole attack detection using energy model in MANETs, 2nd international conference on power and embedded drive control (ICPEDC) (pp. 264–268), IEEE, (2019).
20. Zeng, Y., Qiu, M., Zhu, D., Xue, Z., Xiong, J., & Liu, M : Deepvcml: a deep learning based intrusion detection method in vanet , IEEE 5th intl conference on big data security on cloud (2019).
21. Kaur, J., Singh, T., & Lakhwani: An enhanced approach for attack detection in vanets using adaptive neuro-fuzzy system, ICACTM, IEEE, (2019) :191-197.
22. Chen, Joy Iong Zong, and P. Hengjinda, "Enhanced Dragonfly Algorithm based K-Medoid Clustering Model for VANET." Journal of ISMAC 3, no. 01 (2021): 50-59.

A detailed analysis on spam emails and detection using Machine Learning algorithms

Razia Sulthana A¹[0000–0001–5331–1310], Avani Verma²[0000–0001–8077–4353], and Jaithunbi A K³[0000–0002–2870–6609]

¹ Department of Computing and Mathematical Sciences, University of Greenwich,
Old Royal Naval College, London, United Kingdom, SE10 9LS

razia.sulthana@greenwich.ac.uk

² Department of Computer Science, Birla Institute of Technology & Science, Pilani -
Dubai Campus, United Arab Emirates- 345055

f20190077@dubai.bits-pilani.ac.in

³ Department of Computer Science, RMD Engineering College, Kavaraipettai,
TamilNadu, India, 601206

akj.cse@rmd.ac.in

Abstract. Spam Email is the unwanted junk and solicited email sent in bulk to the receivers, using botnets, spambots, or a network of infected computers. These spam emails can be phishing emails that trick users to get their sensitive information, download malware into the user devices or scam the users stealing confidential data. This paper shows a systematic analysis of spam and its types. It also details the procedure of how the spammers get the email addresses of the receivers. It analyses the problems with spamming. A detailed state of the art on spam filters and the factors that put an email into the spam or ham category is also explained. The paper also discusses spam filtering methods of Gmail, Yahoo, and Outlook. Finally, it brings out several solutions to detect spam using principles of Machine Learning and Data Mining.

Keywords: Spam email · Security breach · Naive bayes · Logistic Regression · Machine Learning

1 Introduction

Spam email is unwanted junk and unsolicited mail sent in bulk to the receiver through an email system like a network of infected computers and botnets. It can also be sent via text messages, phone calls and social media. It can be sent by businesses for commercial reasons. It can also be a malicious attempt to gain access to the user's computer. Since these emails are sent from botnets, they are very difficult to trace and stop.

The links or attachments in the mail might include malicious information. Generally, the hackers use the links or attachments to check the legitimacy of the email addresses or go to malicious websites or downloads which can install the malware in the computer. The users have their email addresses recognised by

spambots. Spambots are automated programs that search the internet for email addresses. Thus, spammers use spambots for generating an email distribution list. Emails are generally sent to millions of users. However, only a small number of users react to these emails.

The number of people communicating with each other online is increasing because of the internet. People depend on emails for general or business related issues. It is a very effective tool for communication as it saves cost and time. In recent years, emails are affected by attacks like spam emails, phishing emails etc. Spam floods receivers inboxes with mimicked messages, or with documents or links which can pass on malware to the device or can trick the receivers to reveal their sensitive information. Thus, spam filters are needed to avoid these. The spam filters should provide high accuracy and have minimal errors and should be efficient too. The objectives of the paper include:

- Understanding the meaning and types of spam emails.
- To analyse the working of spam email filters.
- To discuss case studies of Gmail, Yahoo and Outlook spam filters.
- To provide solutions to detect spam emails using Machine Learning (ML) algorithms (Naive bayes(NB) and Logistic Regression(LoR))..

2 Types of Spam

Spam can be used for the marketing of goods and services or can be malicious. Types of spam are:

1. Phishing Emails - these are sent by cyber attacker to many people which trick people into giving their personal information like bank and credit card details. Phishing is an online scam based on social engineering i.e., malicious activities performed through human interactions. The scammer creates links to click where users can put the sensitive information or download malware into the device.
2. Email Spoofing - the email spoofs an email that resembles the original so that the user can believe the authenticity. The message may be to request payment or to verify/ reset the account or update billing information..
3. Tech Support Scams - these emails explain that the user has some technical issue and provide the phone number of the tech support or a link to click. These emails mimic being a large and reputed company. If the user gives the details of the devices, they can be hacked.
4. Current Event Scams - these emails depend on the current news. For example, during Covid 19, scammers sent messages for work from home that paid in bitcoin or donations to fake organisations.
5. Advance Fee Scams - these scam emails promise a reward, generally financial if some amount of cash is provided in advance for some processing or transfer of money/goods. Once the user pays, they either ask more or disappear.

6. Malware Spam - it is a spam email that delivers malware to the devices. These emails have links or attachments. When the user clicks these, malware like Ransomware, Trojan Horses, Bots, viruses, Spyware etc are downloaded to the device. Generally, the attachments are in the form of a Word document, PowerPoint presentation or PDF file.

Spam filters can be implemented on all layers - firewalls in front of an email server or at Message Transfer Agent (MTA), email server to provide integrated anti-spam and antivirus solution which provides complete email protection at the network level. At the MDA level, spam filters can be installed.

2.1 Understanding how spammers get address

Ways in which spammers get the email addresses:

1. There are thousands of companies that sell CDs containing millions of email addresses. These addresses can be easily formatted and copy-pasted in the 'To' section of the email. The companies get the email addresses from several primary sources like newsgroups and chat rooms. The users generally leave their email addresses in these groups. Software can be used to extract these screen names and email addresses.
2. Secondly, these email addresses can be found on the web. The '@' symbol can be searched on the internet to get the email addresses. This can be done by using a web crawler.
3. Thirdly, the spammer can create sites for winning the lottery in which the user has to type the email address. If they accept receiving the email newsletters, then their email addresses are sold to the spammer.
4. Next, performing a dictionary attack on the email hosting websites can also generate email addresses.

2.2 Problems with spamming

According to Statistica, the global daily spam volume from October 2020 to September 2021 was 1980.58 billion spam emails from 2346.05 billion emails, which accounted for 84.42%. The number of global daily spam volumes in July 2021 had the highest value of 283 billion spam emails out of 336.41 billion emails sent from October 2020 to September 2021. (Table 1 & Figure 1)

Every Internet Service Provider (ISP) pays to use the internet by the purchase of bandwidth. When the volume of spam that is directed to the ISP increases, the bandwidth becomes crowded, and this reduces the speed of internet access. To avoid this situation, the ISP pays to the filtering software or to increase the bandwidth. This expense is often passed to the buyers of ISP.

Some spam emails allow the user to remove themselves from the subscriber list but when the users respond to the email, they verify that their email accounts are active. This might lead to getting more spam emails.

Spam Emails	Total Emails
242.42	286.41
210.54	248.7
140.56	166.38
122.33	144.76
150.93	178.3
138.09	163.87
88.21	104.2
200.24	236.74
249.95	296.81
282.93	336.41
65.5	77.8
88.88	105.67

Table 1. Global daily spam volume from October 2020 to September 2021, Source: Statistica

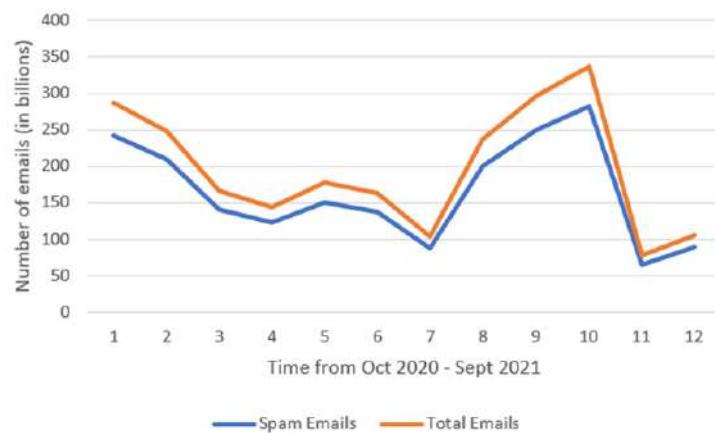


Fig. 1. Global daily spam volume from October 2020 to September 2021. Source: Statistica

2.3 Types of Spam Filters

1. Third-Party or Cloud-based and Gateway spam filters - several companies use cloud based or gateway spam filters for the suspicious inbound and outbound emails. These gateway spam filters are installed on servers. Whereas cloud-based spam filters are run on third-party servers. They are entirely digital. These cloud-based and gateway spam filters provide the network admin to have extra control over in and out traffic of the network.
2. Desktop spam filters - live on the user's device and allow for 1-1 configuration and personalisation. Ex. G-Lock, SpamCombat, Microsoft Smart Screen.
3. Email Service Provider (ESP) built-in spam filters - for Business to Customer or Business to Business senders, Google, Yahoo and Microsoft have inbuilt spam filters and inbox sorting technologies.

2.4 Factors to determine if email is spam or ham

A ham is generally an email that is not a spam.

1. Source Internet Protocol (IP) Address - If a specific IP address has received many complaints in the past, email from that address is more likely to be identified as spam. An email with a poor IP reputation might not be accepted by the server because the IP address is an important factor in delivering an email. Therefore, many companies set up dedicated IP addresses for their users.

If there is no dedicated IP address, the emails will be sent through the marketing automation platform shared IP. Thus, the emails are sent from the same servers as the other customers through these platforms. If any user misbehaves, all other users who are using that platform are affected and the IP reputation decreases.

For dedicated sending IP addresses, the reputation can be checked by using Sender Score, Talos Intelligence and Reputation Authority services. For shared IP, it is important to determine if the email is spam or ham, which will be decided by the domain reputation.

2. Sender's Domain - the ESP look for the sender's originating IP address, sending domain and the sender's alias. If emails of a company's domain are marked as spam, there is a high possibility that these emails are not a priority for the receivers. If ESP labels them as spam, these emails will be missed. The reputation of the domain is not good. Whereas, if the emails are whitelisted, the reputation of the domain is good.
3. Spam Traps - if emails are sent to the spam traps, the domain and IP reputation decreases. If an email account is not used for a specific period of time, the email providers disable it. The ESP might recycle it and convert it to a spam trap. The senders who send spam emails to that account will be fined. Therefore, such email addresses should be removed from the email list. ESP might put fake email addresses so that the bots find them and put them into the mailing list. If the spammers use such mailing lists, they'll be penalised and put on a blacklist.

4. Blacklists - are lists of IP addresses that are owned by known spammers or people who let spammers use their devices. Some of the known blacklists are Return Path Reputation Network Blacklist (RNBL), Sbl.spamhaus.org (SBL), SpamCop (SCBL) etc.
5. Sending Rate - emails can fail to reach the inbox because too many emails are being sent to that server at a time. The gateway filters allow the admin to rate control the bulk email deliveries. Also, if the email is sent to multiple contacts at the same domain, the email might not be delivered. Thus, spreading out of sending email over time can increase the deliverability. The emails can be delivered over a window over time with each recipient getting the email at the predicted time. Throttling follows send-time optimisation which reduces the probability of the email being labelled as bulk delivery.
6. Content - ESP sorts email using Content and IP. Recent spam filter models work on patterns rather than specific words to avoid. The content of the email plays a major role in user engagement. If the email has poorly qualified content, the user might directly mark it as a spam approach and ignore it. Thus, the email should be made keeping the text, images and HTML in mind.
7. Authentication - this is used for ESP to verify the sender's and to prevent spam from reaching the inbox. The emails that don't clear these protocols are considered spam. Types of authentication protocols :
 - Domain Keys Identified Mail (DKIM) - uses EDS to verify if the emails are from the actual domain or spoofed.
 - Sender Policy Framework (SPF) - lets the sender specify the authorisation of the mail servers to send email to the receiver's domain.
 - Domain-based Message Authentication, Reporting, and Conformance (DMARC) - gives options to receivers to handle emails if it fails the SPF and DKIM protocols. It also gives information about the senders from the domain.

2.5 Working of Gmail spam filter

More than 1 billion people use Gmail in a month. Gmail uses several rule-based filters, integrating tensor flow and artificial intelligence into the spam filters. It focuses on IP and Domain Reputation, User Engagement, Content and Sending History.

Gmail looks at both the domain and IP address of the senders to distinguish email between spam and ham. The algorithms check the user response when distinguishing between ham and spam. The content of the email - header, body, link, images etc determine if the email is spam or inbox. The filtering depends on the words that are blacklisted as spam words.

Gmail has a database of blacklisted domains. An email is first checked in this database. If the email or domain is not known, it checks if any links present in the email are malicious or not by comparing them with the database. It will also check for any spelling or grammatical errors by comparing the words in the email with the list of trigger words that are mostly featured in the spam emails.

2.6 Working of Outlook spam filter

Microsoft relies on Sender Reputation Data Network (SRDN) along with engagement, spam traps and complaints to filter spam. SRDN uses a panel of voters from different users to train the spam filters. Emails received can be resent asking the users to vote if the email sent was spam or ham. Higher spam votes will lead the future emails to mostly go to the spam folder. It is harder to lower the complaint rate by sending a large volume of emails using SRDN.

2.7 Working of Yahoo spam filter

Yahoo checks the IP address, Domain, Sender, and Uniform Resource Locator (URL) reputation, along with DKIM and DMARC protocols. If emails have a certain sending rate, and there is a sudden increase in activity, the email can be marked as spam. It follows the same practices as Gmail and Outlook.

3 Literature Review

Based on ML and Data Mining, the following literature review has been done (Table 2): In [16], P. Sharma et al. have focused on the ML [4, 13, 18] by implementing NB and J48 for spam email detection. The dataset is divided into different sets and given as input to each algorithm. Total three experiments are performed and the results obtained are compared in terms of Precision, Recall, Accuracy, F1 score, True Positive (TP) rate, True Negative (TN) rate, False Positive (FP) rate and False Negative (FN) rate. The two experiments are performed using individual Naive Bayes(NB) & J48 algorithms. In [8], P. Pandey et al have examined the ML strategies: NB, Support Vector Machines (SVM) relevance to the issue of spam email detection. Email filtration depends on the data classification approach. For data classification, choosing the best performing classifier is the base. Dataset used is of Ling Spam corpus. Firstly, data is accumulated and represented. Next, dimensionality is reduced by email feature choice.

In [15], M. Sethi et al. have proposed a work that focuses on Natural Language Processing (NLP) [3]. The technique used for detection is the NB and Artificial Neural Network ANN [19]. The steps involved are dataset reading and inspection, text preprocessing, feature set and vectorisation, pipeline. In [7], F. Martino et al. have given information about legitimacy to detect spam. The dataset is from Bruce Guenter Project. The algorithm used are NB, LoR, RFC, SVM. The methodology is first defining classes and features, building a dataset, and using vectors for feeding the classifier. In [9], L. Huang et al. have used NB for the classification of emails. The dataset is from Ling Spam Corpus. The methodology used is to preprocess data, searching common spam keywords. The advantage is that the testing has been carried out on spam encryptions.

In [2], N. Shah et al. have used LoR, k-Nearest Neighbour (K-NN) [14] and Decision Trees (DT) for spam detection. The dataset used is of SMS spam collection. There were 4900 ham samples and 672 spam samples. The advantage

is that the proposed method performance is good as compared with the existing state-of-the-art methods. The limitations is that the research is limited to few algorithms. The work can be improved by comparing more algorithms [23]. In [17], M. Singh et al. have used SVM [12, 1, 20] classifier for SMD. Non-linear SVM is used with two kernel functions - Linear and Gaussian kernel. The dataset is of Spam Assassin Public Corpus. In [5], O.E. Taylor et al. have used SVM and RFC models. The dataset used was of the UCL spam base. The methodology is used to preprocess data and split it into train and test data. It is then checked for accuracy by implementing the algorithms and finally, it is classified into ham/spam. The limitation is that only two algorithms were compared.

Reference	Algorithm	Performance metrics
[16]	Naive Bayes, J48	Precision, Accuracy, F1-score
[8]	Naive Bayes, Support Vector Machines, Logistic Regression	Accuracy
[15]	Naive Bayes, Artificial Neural Network	Precision, Accuracy, F1-score
[7]	Naive Bayes, Logistic Regression, Support Vector Machines	Accuracy
[2]	Naive Bayes	Accuracy

Table 2. Literature Analysis

In [6], A. Naem et al have used k-NN, SVM, Bagging, Boosting and approaches for spam email detection. Thke dataset used is CS-DMC2010 and SpamAssassin. The advantage is that the method gets a low number of selected features and archives a high degree of classification precision. Text mining is used to extract textual signatures in [10]. LoR and DT is applied for spam detection [22, 11, 21].

4 Analysis of Machine Learning algorithms

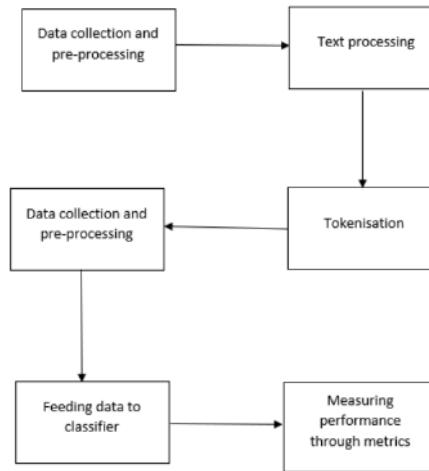
According to Literature Review, hybrid Naive Bayes and Logistic Regression model is most accurate. Naive Bayes is based on Bayes Theorem (Equation 1) with the assumption of strong independence. It is a Probability-Based Classifier. The combinational and frequency values of the dataset are calculated under a probability set. The class that is nearer to the rear end is picked by the classifier.

$$P\left(\frac{a}{b}\right) = \frac{P\left(\frac{b}{a}\right)P(a)}{P(b)} \quad (1)$$

b = set of feature vectors

a = class variable

P(a/b) = posterior probability that depends on the likelihood of attribute value

**Fig. 2.** Architecture Diagram

of class $P(b/a)$

$P(a)$ = prior probability

$P(b)$ = probability of known attribute value

Logistic is an analysis method to model the data and explain the relation between the Binary Response Variable and Explanatory Variable. The result is the probability of assigning a value to a particular class, which is in the range of 0 to 1.

1. Initially, data collection and pre-processing is done by removing undefined values, gaps and duplicates. This helps to reduce errors and improve the quality of classification.
2. Secondly, text processing is done to remove unwanted noise and characters like punctuation and numbers. This is done by converting all letters to lowercase, deleting numbers, and removing punctuation marks and stop words like prepositions and pronouns etc.
3. Tokenisation is performed by splitting sentences into words separated by a comma.
4. Finally, the quality of the classification is accessed. Model training is performed by the metrics namely Accuracy, Precision (Figure 2)

The dataset used for spam filtering is Ling spam dataset. It includes 1000 emails. The dataset is divided into 80:20 split ratio and are subjected to naive bayes and logistic regression. The training and testing results are shown in Table 3 and Table 4 respectively.

The results are shown in Figure 3. The precision, accuracy, F1 score of Naive bayes and Logistic Regression is recorded. The training and testing results shows a trivial difference which is because of the behavior of the model to the test data

Algorithm	Precision	Accuracy	F1-score
Naive Bayes	96.5	97.3	96.8
Logistic Regression	97.1	96.2	97.04

Table 3. Training Results for the Naive Bayes and Logistic Regression

Algorithm	Precision	Accuracy	F1-score
Naive Bayes	95.5	94.1	95.02
Logistic Regression	93.6	94.5	93.7

Table 4. Testing Results for the Naive Bayes and Logistic Regression

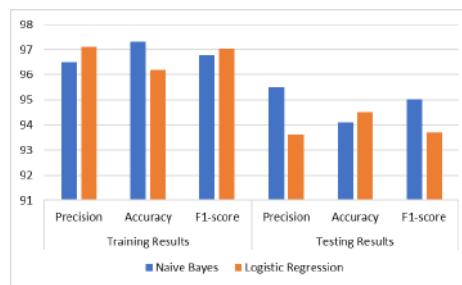
is unpredictable. However, the performance of the Naive bayes system is slightly better than logistic regressin in testing data and is vice versa in test data.

5 Conclusion

Spam email detection helps to detect the unwanted emails and threats. Many researchers are working in this field to find out the best classifier that is efficient and provides high accuracy in detecting spam emails and filtering them. Gmail, Outlook, Yahoo and other email service providers use ML algorithms integrated with AI and Data Mining to build their spam filter models. For ML based methods, NB and LoR prove to be the most efficient algorithms used to detect spam email. For Data Mining, the best method is RT proved to be the most accurate method with high accuracy.

References

- Chauhan, A., Agarwal, A., Sulthana, R.: Genetic algorithm and ensemble learning aided text classification using support vector machines. International Journal of Advanced Computer Science and Applications **12**(8) (2021)

**Fig. 3.** Result analysis

2. GuangJun, L., Nazir, S., Khan, H.U., Haq, A.U.: Spam detection approach for secure mobile message communication using machine learning algorithms. *Security and Communication Networks* **2020** (2020)
3. Kontsewaya, Y., Antonov, E., Artamonov, A.: Evaluating the effectiveness of machine learning methods for spam detection. *Procedia Computer Science* **190**, 479–486 (2021)
4. Mathur, A., Sultana, R.: A study of machine learning algorithms in speech recognition and language identification system. In: *Innovations in Computer Science and Engineering*, pp. 503–513. Springer (2021)
5. Naem, A.A., Ghali, N.I., Saleh, A.A.: Antlion optimization and boosting classifier for spam email detection. *Future Computing and Informatics Journal* **3**(2), 436–442 (2018)
6. Naem, A.A., Ghali, N.I., Saleh, A.A.: Antlion optimization and boosting classifier for spam email detection. *Future Computing and Informatics Journal* **3**(2), 436–442 (2018)
7. Nidhya, M., Jayanthi, L., Sekar, R., Jeyabharathi, J., Poonam, M.: Analysis of machine learning algorithms for spam filtering. *Annals of the Romanian Society for Cell Biology* pp. 3469–3476 (2021)
8. Pandey, P., Agrawal, C., Ansari, T.N.: A hybrid algorithm for malicious spam detection in email through machine learning. *Int J Appl Eng Res* **13**(24), 16971–16979 (2018)
9. Peng, W., Huang, L., Jia, J., Ingram, E.: Enhancing the naive bayes spam filter through intelligent text modification detection. In: *2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE)*. pp. 849–854. IEEE (2018)
10. Qian, F., Pathak, A., Hu, Y.C., Mao, Z.M., Xie, Y.: A case for unsupervised-learning-based spam filtering. *ACM SIGMETRICS performance evaluation review* **38**(1), 367–368 (2010)
11. Rathi, M., Pareek, V.: Spam mail detection through data mining-a comparative performance analysis. *International Journal of Modern Education & Computer Science* **5**(12) (2013)
12. Razia, S.A., Pranav, R.: Predicting the import and export of commodities using support vector regression and long short-term prediction models. *International Journal of Computing and Digital Systems* **11**(1), 635–648 (2022)
13. Razia Sulthana, A., Mathur, A.: A state of art of machine learning algorithms applied over language identification and speech recognition models. In: *International Virtual Conference on Industry 4.0*. pp. 123–132. Springer (2021)
14. Sajedi, H., Parast, G.Z., Akbari, F.: Sms spam filtering using machine learning techniques: a survey. *Machine Learning Research* **1**(1), 1–14 (2016)
15. Sethi, M., Chandra, S., Chaudhary, V.: Email spam detection using machine learning and neural networks. *Int. Res. J. Eng. Technol* **8**, 349–355 (2021)
16. Sharma, P., Bhardwaj, U.: Machine learning based spam e-mail detection. *International Journal of Intelligent Engineering and Systems* **11**(3), 1–10 (2018)
17. Singh, M., Pamula, R., et al.: Email spam classification by support vector machine. In: *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*. pp. 878–882. IEEE (2018)
18. Sulthana, A.R., Jaithunbi, A.: Varying combination of feature extraction and modified support vector machines based prediction of myocardial infarction. *Evolving Systems* pp. 1–18 (2022)

19. Sulhana, A.R., Jaithunbi, A., Ramesh, L.S.: Sentiment analysis in twitter data using data analytic techniques for predictive modelling. In: Journal of Physics: Conference Series. vol. 1000, p. 012130. IOP Publishing (2018)
20. Trivedi, S.K.: A study of machine learning classifiers for spam detection. In: 2016 4th international symposium on computational and business intelligence (ISCB). pp. 176–180. IEEE (2016)
21. Vivekanandam, B., et al.: Spam email classification by hybrid feature selection with advanced machine learning algorithm—future perspective. *Journal of Soft Computing Paradigm* **4**(2), 58–68 (2022)
22. Wijaya, A., Bisri, A.: Hybrid decision tree and logistic regression classifier for email spam detection. In: 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE). pp. 1–4. IEEE (2016)
23. Yasin, A., Abuhasan, A.: An intelligent classification model for phishing email detection. arXiv preprint arXiv:1608.02196 (2016)

Advanced Encryption Standard based Encryption for Secured Transmission of Data in Cognitive Radio with Multi-Channels

Kiran P More¹ and RajendrakumarA Patil²

¹Dept.of Electronics and Telecommunication Engineering

College of Engineering Pune

Pune,India

kiran.more@.edu

²Dept.of Electronics and Telecommunication Engineering

College Of Engineering Pune

Pune,India

rap.extc@coep.ac.in

Abstract: CRN is hypes as a powerful tool for advancing 5G networks that can significantly increase SE by allowing unlicensed users to access the inactive licensed spectra without interfering with licensed PUs. Moreover, CRN becomes more challenging due to the difficulty in accessing and detecting the channel. In this study, a Self Upgraded Spider Monkey Optimization (SU-SMO) based LSTM algorithm is used to predict the channel state. Additionally, it aims to carry out secure communication over the foreseen spectrum channels. The Advanced Encryption Standard (AES) guarantees the security level of data frames to enable safe communication. Finally, an analysis was conducted to show how the model could be improved.

Keywords: SU-SMO (Self Upgraded Spider Monkey Optimization); Spectrum efficiency; Cognitive Radio; LSTM(Long short-term memory); Advanced Encryption Standard Concept.

Nomenclature

Acronym	Description
MSS	Multi-Band Spectrum-Sensing
M-AES	Modified Advanced Encryption Standard
VANET	Vehicular Network
CRSN	Cognitive Radio Sensor Network
ACD	Auto-correlation based Detector
ROC	Receiver Operating Characteristics
CR	Cognitive Radio
LSTM	Long Short-Term Memory
Pus	Primary Users
DBN	Deep Belief Network
MHTP	Multichannel Hidden Terminal Problem
ProMAC	Proactive Medium Access Control protocol
APS-MAC	Adaptive Preamble Sampling-based MAC

CRV	Cognitive radio for VANET
CTS	Clear To Send
MME	Maximum-Minimum Eigen value detectors
DCNN	Deep Convolutional Neural Network
SNR	Signal-To-Noise Ratio
CS-GOA	Cuckoo Search-Grasshopper Optimization Algorithm
SM	Spider Monkey
RARE	spectRum Aware cRoss-layEr
MPC	Model Predictive Control
SUs	Secondary Users
MA	Multiple Access
MAC	Medium Access Control
SVD	Singular Value based Detector
SDR	Software Defined Radio
CFD	Cyclostationary Feature based Detector
RTS	Request To Send
CRN	Cognitive radio Networks
FMAC	Fairness-based MAC
ED	Energy Detector
SE	Spectrum Efficiency
ECC	Elliptic Curve Cryptography
RSA	Rivest-Shamir-Adleman
DSA	Dynamic Spectrum Access

1. Introduction

Currently, the fixed spectrum distribution technique is widely used. Though, the majority of available bandwidth is not used as the requirements of bandwidth by apps or devices are constantly changing [1] [2]. The spectrum remains inactive while a licensed band is not used by a licensed user (also known as PU) since it cannot be used by several other unlicensed clients [3] [4]. According to the measurements, a fixed allocation scheme's usual value of spectrum utilization ranges from 15% to 85% [5] [6]. These current spectrum efficiency problems are resolved using a technique known as DSA. Also, the method called CRN might be used to enable the DSA [7] [8].

In the DSA technique, the primary user's uninterrupted operation is maintained while the licensed band is made accessible to unlicensed users. SUs are devices with CR technology capabilities [9]. DSA comes in two varieties, including spectrum overlay and underlay. SU maximum transmission parameters allow SUs to transmit beneath the conflicting temperature limit of PUs in spectrum underlay. Spectrum overlay, however, does not apply any power constraints on SUs [10]. SUs are able to identify and utilize available channels thanks to spectrum overlay. An SDR called CRN has the ability to modify its transmission/reception restrictions depending on the state of the surrounding environment. The cognitive cycle is a factor in CRN functions. The following list summarizes this work's main contribution:

- Presents the SU-SMO-based LSTM technique for detecting the presence of channel states.

- Suggests the usage of AES to maintain the data securely across the licensed spectrum channel.

The remaining work is organised as follows: Section 2 examined a review on a related subject. The system modelling is shown in Section 3, and the suggested MPC channel allocation procedure is shown in Section 4. Section 5 describes the suggested AES protocol for secure data transport. The results and conclusion are explained in Sections 5 and 6.

2. Literature Survey

2.1 Related Works

In 2018, Nafees *et al.* [11] created a technique that divides a network into several groups as well as the layout of the clusters appears to be a bigger challenge for edge biclique. Moreover, super-frame architecture was suggested to maintain network dependability. A delay-aware protocol from the accepted RARE was still a problem with the weighted graph. It was verified that RARE-modified clusters oriented on node mobility and spectrum availability.

In 2018, Li and Han [12] have developed a plan to reduce congestion by allocating time and channels as efficiently as possible. In contrast, the handshaking process for establishing a data link received very little attention. With this strategy, a novel configuration that incorporated CRAHN features was demonstrated for CTS and RTS frames. As a result, an FMAC method was also made available that dealt with the distribution of data links among SUs.

In 2018, Manyet *et al.* [13] created an APS-MAC framework that enabled the evaluation of opportunistic spectra and addressed CRSN energy efficiency. The proposed layout that enabled the duty cycle on CRSN had preamble sampling as a key component. Additionally, because each CRSN sets its own wake-up and sleep schedule through sampling, it is possible for them to remain in a drowsy state indefinitely.

In 2020, Moayadet *et al.* [14] emphasized the multi-stage channel assignment problem as a spectrum accessing challenge. The goal was to increase the acquired sum rate on CR-IoT nodes in order to increase the overall network throughput. An innovative resource-oriented channel allocation approach that provided appropriate use of the provided time-frequency unit was primarily used.

In 2016, Sathya *et al.* [15] proposed an MPC-focused ProMAC in CRN for SUs, the first proMAC model to be created. When compared to existing systems described in the literature, ProMAC deployment with fixed SU and PU counts led to enhanced channel usage as well as the best sensing lag & back-off rate.

Table:1 Examining conventional CRV protocols

Author [citation]	Adopted model	Features	Challenges
Nafees <i>et al.</i> (2018)[11]	RARE protocol	Does not cluster as much. Delivers reliable communication	Finding neighbours is not possible.
Li and Han (2018) [12]	FMAC protocol	Enhanced link distribution An extended window for rendezvous success.	Disorders develop once data allocation is finished on schedule.
Manyiet <i>et al.</i> (2018) [13]	APS-MAC protocol	Greater energy efficiency High throughput.	Threats cause injustice to happen.
Moayadet <i>et al.</i> (2020) [14]	MAC protocol	Excellent spectral efficiency High throughput	Only a fixed basic rate is supported by each channel.
Stahyaet <i>et al.</i> (2016) [15]	ProMAC	Provides efficient channel assignment.	Difficult assessment procedure.

Various methods have been focused on predicting the channel states (busy or idle) using prediction models. However, there exist issues such as identifying neighbours is impossible, problems arise after data allocation is completed on schedule, threats result in injustice, each channel only supports a fixed basic rate, and the assessment process is challenging. Therefore, in order to rectify the abovementioned issues, this study develops a novel model for detecting the presence of channel states. The proposed model is described in the following section.

3. Proposed MPC Protocol for Channel Allocation

MPC model [16] includes five major steps. Step I of the MPC model explains the variable boundaries, while Step II predicts the values of the prediction variables. Step III conducts system construction and takes decisions based on the predictions. Step IV is concerned with using Step II's predictions to generate the values of variables. The results of Steps IV and III are evaluated in Step V. When SU needs a channel in a VANET, the MPC architecture is then implemented like a control system. The flow process of ProMAC based framework is shown in Fig. 2

The major goal is to address the following issue:

- ✚ The ProMAC predicts the condition of transmission (active or idle) of PU for subsequent ST time slots using the PU channel list and its prior Tr transmission states.
 - ✚ Additionally, the busy or idle state of the PU channel is determined based on whether the associated PU is transferring or not.
 - ✚ Considering E channels connected to PUs, which are indicated by $\{F_1, F_2, \dots, F_E\}$.
- Here, $h_j^i = F_i$ location in j time slot
- ✚ A history polynomial $HisP_i$ connected to F_i for the designated tp time period is shown in Eq. (1).

$$HisP^i(x) = \sum_{j=tp-h}^{tp-1} h_j^i * x^{j-tp+h}, \quad h_j \in \{0,1\}$$

$$h_j^i = \begin{cases} 1, & \text{if } F_i \text{ is busy at time slot } tp \\ 0, & \text{Otherwise} \end{cases} \quad (1)$$

The work that has been developed focuses on secure data transfer on the expected channel conditions. Additionally, as a further development, this work uses SU-SMO based LSTM to predict the state of transfer.

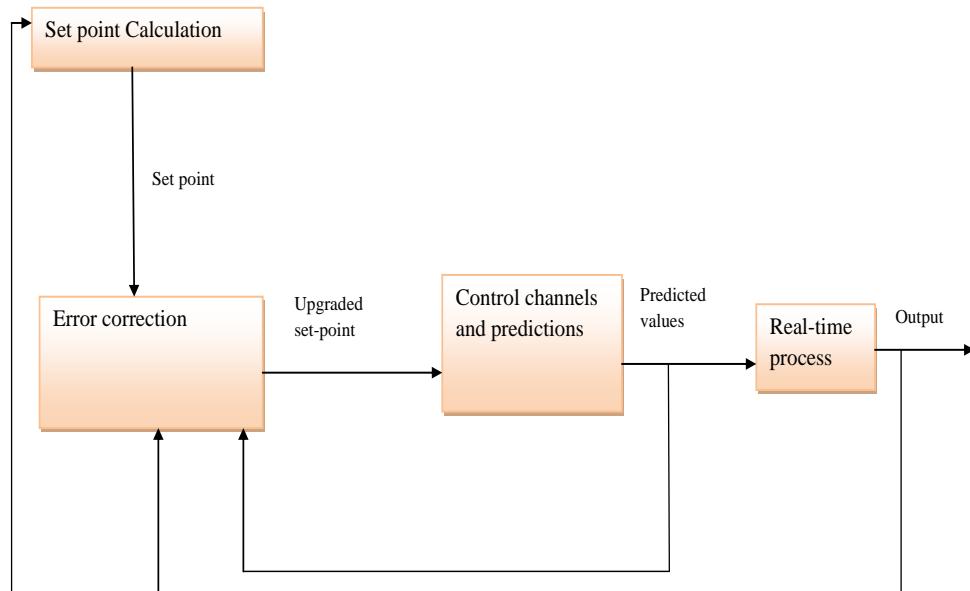


Fig. 1 Flow diagram for ProMAC

3.1 Optimized LSTM classifier

Three units, such as the forget gate, an input gate, as well as the output gate, make up every LSTM cell. Letting L and Q stand for hidden and cell state, respectively, in that order. Here, $(G_t, Q_{t-1}, L_{t-1}), (M_t, C_t)$ = input, output layers.

An output, input, & forgets gate at a time t , signifies I_t, l_t, o_t , respectively. o_t is mostly used by LSTM to categorize the information into irrelevant categories. o_t is given in Eq. (2).

$$o_t = \diamond(T_{lo}G_t + O_{lo} + T_{L_o}L_{t-1} + O_{Lo}) \quad (2)$$

Where, $(T_{Lo}, O_{Lo}), (T_{lo}, O_{lo})$ = weight and bias parameters to mapping hidden layers & forget gate's input layers, \diamond = activation function.

LSTM uses the input gate as demonstrated in Equations (3) and (4) and (5).

Here, $(T_{lX}, O_{lX}), (T_{lx}, O_{lx})$ = cell gate's hidden & input layers are mapped using the weight and bias parameter. $(J_{lI}, B_{lI}), (J_{li}, B_{li})$ = mapping hidden & input layers to l_t using weight and bias parameter. tanh has traditionally been used as an activation function. This approach is unique in that it uses leaky ReLu as the activation function rather than tanh . Additionally, as shown in Equations (6) and Eq. (7), the LSTM cell acquires the output hidden layer via the output gate.

$$X_t = \text{leakyReLu}(T_{lX}G_t + O_{lX} + T_{lX}L_{t-1} + O_{lX}) \quad (3)$$

$$l_t = \diamond(T_{li}G_t + O_{li} + T_{lI}L_{t-1} + O_{lI}) \quad (4)$$

$$Q_t = o_t Q_{t-1} + l_t X_t \quad (5)$$

$$I_t = \diamond(T_{li}G_t + O_{li} + T_{lI}L_{t-1} + O_{lI}) \quad (6)$$

$$L_t = I_t \text{leakyReLu}(Q_t) \quad (7)$$

Here, $(T_{lI}, O_{lI}), (T_{li}, O_{li})$ = weight and bias to transfer the input layer's hidden values to I_t . Also $T_{lo} + T_{Lo} + T_{lX} + T_{lI} + T_{lx} = T$ were fine-tuned by SU-SMO approach.

Solution encoding:As noted previously, the SU-SMO technique is used to select the LSTM weights denoted by (T) . The depiction for solutions is shown in Fig. 2.

Where N = LSTM weights total. The proposed work's objective is to reduce the error (err) as given in Eq. (8).

$$obj = \min(err) \quad (8)$$



Fig:2 Solution encoding

3.2 SU-SMO Algorithm

This study proposes an innovative and effective strategy employing the Spider Monkey Optimization method to solve a complicated optimization issue. The newly suggested method is self upgraded in that it changes the role of the local leader in accordance with its existing situation.

The suggested SU-SMO algorithm operates in the way described below:

- ✓ **Initialization:** During the initial phase, SMO generates a uniformly distributed beginning swarm of M spider monkeys. Here, $SiM_i = i^{th}$ SM in swarm. SiM_i was initialized as per eq. (9):

$$SiM_{ij} = SiM_{\min j} + Z \times (SiM_{\max j} - SiM_{\min j}) \quad (9)$$

Here, $SiM_{\min j}$, $SiM_{\max j}$ = lower/upper bound Search region, Z = random numbers with an even distribution

Modified Local Leader Phase (LLP): In eq. (10), the LLP position update solution is shown.

$$SiM_{new_{ij}} = SiM_{\min j} + Z_1 \times (lol_{kj} - SiM_{ij}) + Z_2 \times (SiM_{rj} - SiM_{ij}) \quad (10)$$

Here, SiM_{ij} = i^{th} SM's j^{th} dimension, lol_{sj} = s^{th} set local leader's j^{th} dimension, SiM_{rj} = SM's j^{th} dimension which was randomly selected.

As per the upgraded logic, LLP position update is carried out as in eq. (11)

$$SiM_{new_{ij}} = SiM_{\min j} + Z_1 \times (lol_{kj} - SiM_{ij}) + Z_2 \times (SiM_{rj} - SiM_{ij}) + O * (SiM_{best} - SiM_{worst}) \quad (11)$$

Here, $O = \frac{O_1 + O_2}{2}$, which was adaptive search factor

Where,

$$O_1 = e + (f - e) * \delta, e = -1.2$$

$$O_2 = f + (f - e) * \delta, f = 1.2$$

$\delta = 0.618$, which was a scaling factor

Modified Global Leader Phase (GLP): It is possible to assess fitness F_i from objective function fc_i . The accompanying equation (12) is used to calculate i^{th} SM's likelihood of being selected for the GLP if F_i is its fitness:

$$prb_i = \frac{Fn_i}{\sum_{i=1}^N Fn_i} \quad (12)$$

Here, prb_i = selection probability. The standard position update formula is displayed in this phase as eq. (13):

$$SiM_{new_{ij}} = SiM_{ij} + Z_1 \times (gl_j - SiM_{ij}) + Z_2 \times (SiM_{rj} - SiM_{ij}) \quad (13)$$

As per the upgraded model, SU-SMO, the GLP update is carried out in eq. (14).

$$\begin{aligned} SiM_{new_{ij}} = SiM_{ij} + Z_1 \times (gl_j - SiM_{ij}) + \\ Z_2 \times (SiM_{rj} - SiM_{ij}) * c \end{aligned} \quad (14)$$

Where, c = inertia weight, evaluated in eq. (15)

$$c = c_{\max} - \frac{c_{\max} - c_{\min}}{I_{\max}} * I \quad (15)$$

Where,

$c_{\max} = 1$, $c_{\min} = 0$, I = current iteration, I_{\max} = maximum iteration

4. Suggested Secure Data Transmission using AES Protocol

4.1 Channel Sensing Phase

The channel sensing step is responsible for identifying the band gaps that will be used in the communication between Tx-Rx pairs. It is assumed that the SUs employ energy recognition to sense spectra. ProMAC combines channel sensing on favor of the MAC layer to receive the most recent sensing data before starting transmission [17]. Be aware that due to hardware constraints, it may not be possible to examine all available data channels within a given sensing time. Using both the DCRN and CCRN systems, the proposed model identified a collection of idle channels that were sensed voluntarily.

4.2 Contention Phase

Depending on the MA mechanism described in "IEEE 802.11 DCF,"[18] the SUs compete for and avoid an inactive PU channel throughout this stage. Each Tx-Rx pair reserves a maximum of one data channel. The MHTP issue is handled as a result of the implementation of the CTS / RTS mechanism for generating communication as described in IEEE 802.11 DCF. The projections produced by the MPC-oriented ProMAC system are being tested at this level. ProMAC is specifically given the condition of each sensing channel as input. After detecting PU state channel, data transmission is decided according to the detected state

(busy or idle). As a consequence, the ProMAC learns whether its predictions are consistent with the actual situation and advances toward increased predictive accuracy in the future. In the conclusion, it lowers the likelihood that similar disputes may arise during the following beacon time period.

4.3 Data transmission Phase

Every transmitter SU node that successfully retain PU channel during the contention stage starts to transmit data to its anticipated receiver node using that channel during the data transmission stage. As every Tx-Rx pair is communicating on a different channel, every such data transfer occurs concurrently. As a result, it is presumptive that a single packet or many packets may be broadcast and that the datagram of a MAC packet has a predetermined size. If a few Tx-Rx pairs encounter PU intervention at any point during this phase, the broadcasting will be stopped as well as the SU node will back off. A smash with PU could occur in this step if the Tx-Rx pair has any sensing errors throughout the sensing stage. In the following beacon period, the collision-affected Tx-Rx couples would proceed through the contention & sensing phases. As a result, only one beacon time is wasted after collisions.

The proposed work uses AES to transmit data securely.

4.4 AES

AES is one of the encryption methods used to safeguard internet data from harmful threats. It is the most reliable security protocol because it is used in both hardware and software. For encryption, it employs longer key sizes, including 128, 192, and 256 bits. As a result, the AES algorithm is more secure against hackers. This makes it extremely challenging to hack, making it a really safe protocol. AES has four modifications that quickly interrupt plain text in order to increase security. Due to its lower expenses, it might be made simpler on any paradigm as well. AES uses a key size of 128, 192, and 256 bits and a block size of 128 bits, with related cycle counts of 10, 12, and 14, respectively. Mix Columns & Add Round Key is two of the four types of transformations that are included. AES is speedy in both hardware and software and is dependent on a design approach called as a substitution-permutation system, which combines both substitution as well as permutation.

4.5 Encryption

The steps that must be taken in the encryption method:

- ❖ Divide a 128-bit key into 8 equally-sized halves.
- ❖ Evaluate Ua' , Ub' as per Eq. (16), Eq. (17) & Eq. (18).

$$Ua' = \left(f - a_0^2 + b_0 \right) + \left(f - a_2^2 + b_2 \right) + \left(f - a_4^2 + b_4 \right) + \left(f - a_6^2 + b_6 \right) \quad (16)$$

$$Uz' = 4f - \sum_{n=0}^6 \left(a_{2n}^2 - b_{2n} \right) \quad (17)$$

$$Ub' = g \sum_{n=0}^7 a_{2n+1} \quad (18)$$

- ❖ To cause confusion, the 128 bit key (Ua') would be XORed well with 128 bit block data.
- ❖ Mix the output rows
- ❖ Mix output columns
- ❖ Mix columnoutput was XORed with Ub' .
- ❖ As a result, cipher text is formed as per Eq. (19), and Eq. (20).

$$Cipher(Cip) = (((pl \oplus Ua')RW)CL) \oplus Ub' \quad (19)$$

$$Plain(Pl) = (((Cip \oplus Ub')CL)RW) \oplus Ua' \quad (20)$$

Here,

Pl = plain text

Cip = cipher text

RW = rows

CL = mixcolumn

- ❖ The first key would be the output of Ua' , which is XORed to 128 bit data blocks.
- ❖ Both mix rows and mix column steps of the XORed output are performed successively.
- ❖ Therefore, Ua' and Ub' are generated using the Henon map.

4.6 Decryption

- ❖ The steps that are taken in the decryption algorithm is as follows:
- ❖ Divide cipher as 128 bit
- ❖ Split secret key as 8 divisions
- ❖ Form Ua' , Ub' keys
- ❖ Use key Ub' to reverse XOR the cipher
- ❖ Inverse mix rows as output
- ❖ Inverse mix column as output
- ❖ As seen in Eq. (20), plain text is generated.

5. Results and Discussion

5.1 Simulation Procedure

The proposed Self Upgraded Spider Monkey Optimization (SU-SMO) model was executed in NS-2 and the outcomes of the experiments were examined. Here, two experiments had been conducted; the first experiment incorporate with 100 nodes and the second experiment incorporate with 200 nodes. Furthermore, the

suggested model's performance analysis was contrasted with those of more traditional approaches, including Deep Convolutional Neural Network (DCNN), Butterfly Optimization Algorithm (BOA), Deep Belief Network (DBN), Seagull Optimization Algorithm (SOA), Rock Hyraxes Swarm Optimization (RHSO), Spider Monkey Optimization based Long Short Term Memory (SMO-based LSTM), and Shark Smell Optimization Algorithm (SSOA), in terms of average backoff, sensing delay, throughput, and channel utilization. The simulation outcomes for the suggested model were displayed in Fig. 3. The simulation parameters implemented in this experiment are depicted in Table.2.

Table.2 Simulation parameters implemented for evaluation

Parameters	Values
Number of Nodes	50
Number of Rounds	100
Simulation Time	15
Number of Nodes	50
Number of Rounds	100

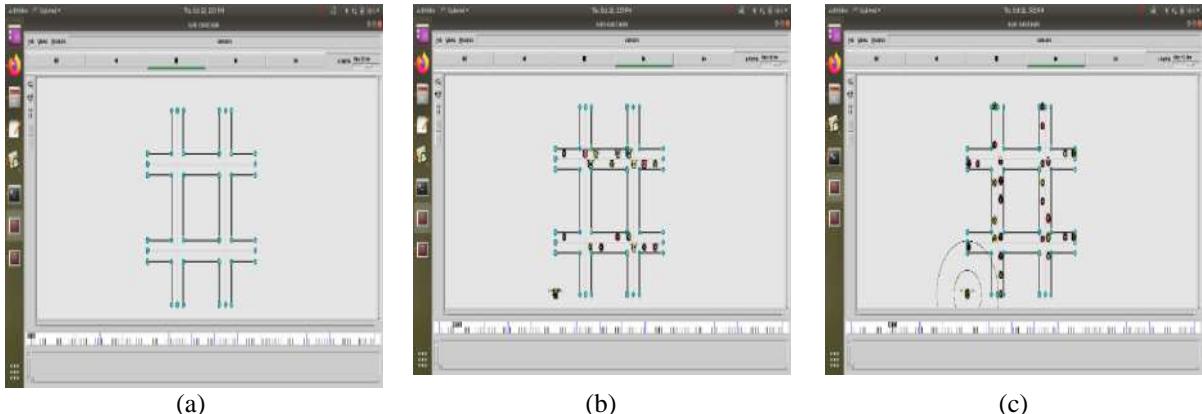


Fig.3 Sample picture of (a) Starting stage (b) and (c) Mid stage while communicating with SU

5.2 Performance Evaluation with regard to average Backoff

Fig 4 provides a more detailed illustration of the suggested SU-SMO model's effectiveness compared to the DCNN, BOA, DBN, SOA, RHSO, and SMO-based LSTM and SSOA in terms of average backoff. The average backoff obtained by the suggested model under 80 PUs is (\sim)0.375, outperforming the values attained by the following models: DCNN (0.436), BOA (0.474), DBN (0.416), SOA (0.429), RHSO (0.415), SMO based LSTM (0.371), and SSOA (0.398), respectively. Considering 90 SUs, BOA and SOA exhibit the lowest performance (0.482 and 0.445 of average backoff), meanwhile the suggested model accomplished 0.379 of average backoff. The proposed work was then performed at 100 SUs with a minimal average backoff of 0.394, although this is substantially

greater than the extant approaches like BOA=0.512, SOA=0.489, and DCNN=0.458.

In a similar manner, the proposed model accomplished with low average backoff in experiment 2 is portrayed in fig 4(b). This follows the same procedure as experiment 1. The suggested technique achieved the required average backoff of 0.345, 0.348, and 0.389 in the 20, 40, and 60 PUs. The proposed work produces 0.386 of average backoff under the final analysis of 100 SUs, which is significantly better than the maximum average backoff that is held by the conventional models, CNN (0.463), BOA (0.549), DBN (0.432), SOA (0.433), RHSO (0.458), SMO based LSTM (0.465), and SSOA (0.459), respectively. The figures make it abundantly clear that the suggested technique has demonstrated superior results to the other models, in recent times. Therefore, the proposed SU-SMO approach is unique in that it is thought to be the optimum strategy for secure data transfer in CRNS.

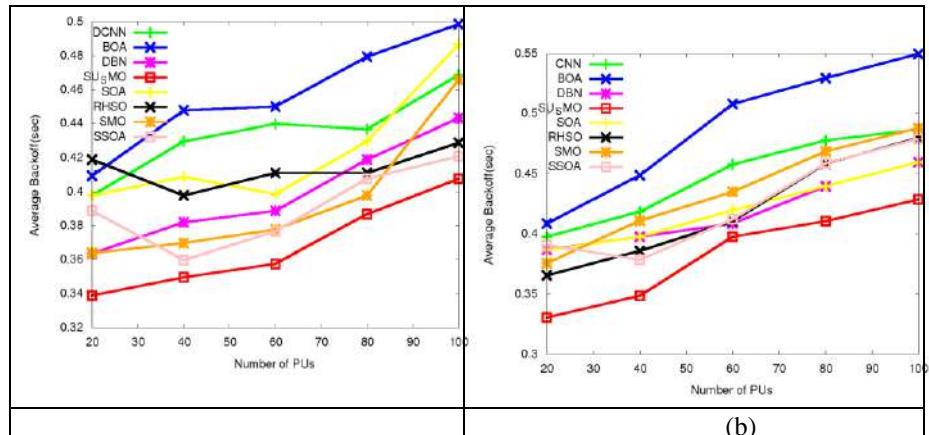


Fig:4 Evaluation on Average Backoff of the proposed SU-SMO model in contrast to the conventional approaches by varying the PUs a) Experiment 1 b) Experiment 2

5.3 Performance Evaluation with regard to Sensing Delay

Fig. 5 compares the performance of the suggested SU-SMO work to the established methods with respect to sensing delay. When analyzing the results of the sensing delay, it is fascinating to observe that the proposed task was accomplished with less sensing delay than the traditional ways. The CNN, BOA, and DBN have sensing delays of 0.539, 0.621, and 0.556 respectively, which is significantly higher than the suggested task (0.362 of sensing delay). The enhanced sensing delay results for the SOA, DBN, and RHSO models are 0.211, 0.234, and 0.236, respectively. Likewise, the CNN and SMO-based LSTM model exhibits considerable effectiveness having sensing delays of 0.254 and 0.268.

However, the presented technique has exceeded the compared techniques by establishing a minimal sensing delay of 0.134 (under the 100 PUs).

A review of the comparison result analysis is shown in fig 5(b), to ensure the safe transfer of data for the suggested model. The findings indicate that the SSOA, CNN, and LSTM technique based on SMO, has produced ineffectual results with sensing delays of 0.238, 0.217, and 0.198, whilst the proposed recorded the lowest sensing delay of 0.121. The proposed SU-SMO model proved that it has a minimal sensing delay than the conventional approaches. Altogether, it serves as a suitable tool for CRNS safe data transmission.

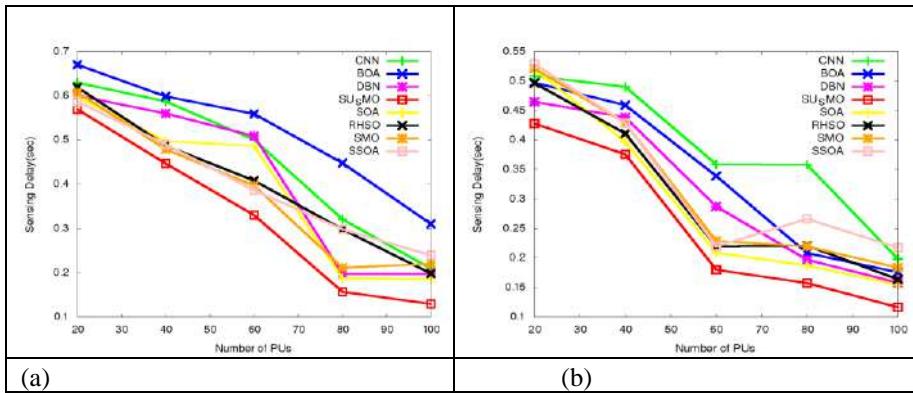


Fig:5 Evaluation on Sensing Delay of the proposed SU-SMO model in contrast to the conventional approaches by varying the PUs a) Experiment 1 b) Experiment 2

5.4 Performance Evaluation with regard to Throughput

The proposed SU-SMO model's throughput analysis for both experiments is presented in Fig. 6(a), and 6(b), in comparison to the traditional approaches. According to the examination of the suggested technique, the throughput value decreases as the number of PUs rises. However, even at 100 PUs, the proposed work performed better than the alternative methods. Additionally, for 20 SUs, the suggested model has a throughput of 89.13 while other existing approaches, such as SMO-based LSTM, RHSO, and CNN, get throughputs of 88.54, 97.95, and 87.32. Furthermore, analysis has been taken and it is discovered that the proposed approach gained a throughput of 84.63, in the 100 SUs, this is extremely high than the throughput of other existing works.

In Fig. 6(b), the conventional models, such as SMO-based LSTM, DBN, and RHSO, achieved moderately improved results with throughputs of 85.67, 85.28, and 84.92, under the 80 PUs. Nevertheless, the adopted work has reached the highest throughput (87.84) than the other models. At the 100 PUs, the proposed model gained a throughput of 86.91, outperforming the DBN (85.32) and SSOA (85.16). The exhibited outcome affirms the reliability of the suggested SU-SMO work for secured data transmission in CRNS at maximum throughput.

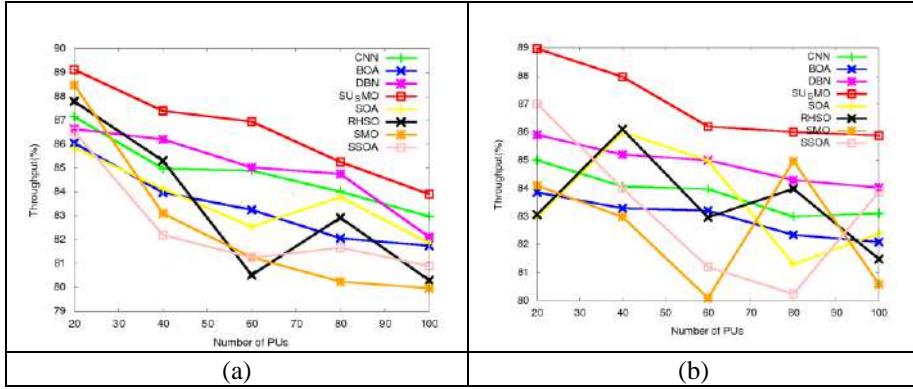


Fig:6 Evaluation on Throughput of the proposed SU-SMO model in contrast to the conventional approaches by varying the PUs a) Experiment 1 b) Experiment2

5.5 Performance Evaluation with regard to Channel Utilization

The comparison between the suggested SU-SMO model and conventional models based on channel usage is illustrated in Fig. 7. Overall analysis shows that the suggested model outperformed other existing strategies in terms of high channel utilization. For the 60 PUs, the proposed method gained with highest channel utilization as 88.27%, therefore it is preferable to the other extant models such as RSMO=88.27%, SSOA=86.92%, DBN=86.18%, CNN=85.68%, SOA=85.12%, SMO based LSTM=84.21% and SSOA=83.34%, respectively. Similar to this, under the 100 PUs, the minimal channel is consumed by the BOA method (83.14%), after that SSOA (84.08%) and SOA (84.81%), though the proposed method acquired the channel utilization as 89.23%.

Simultaneously, when examining fig. 7(b), the suggested model once more showed that it has the capacity to safely transfer the data in CRNS. The channel utilization under the 20, 40, 60, and 80 PUs of the suggested model is 90.98%, 90.32%, 92%, and 91.48%, respectively. Furthermore, with 100 PUs, the RSMO and SOA have recorded channel usage of 90.67% and 98.33%, respectively, but the proposed work recorded the maximum channel usage of 91.26%. Moreover, the suggested SU-SMO method is evaluated against other methods, and the results demonstrate that our adopted method has a higher channel usage.

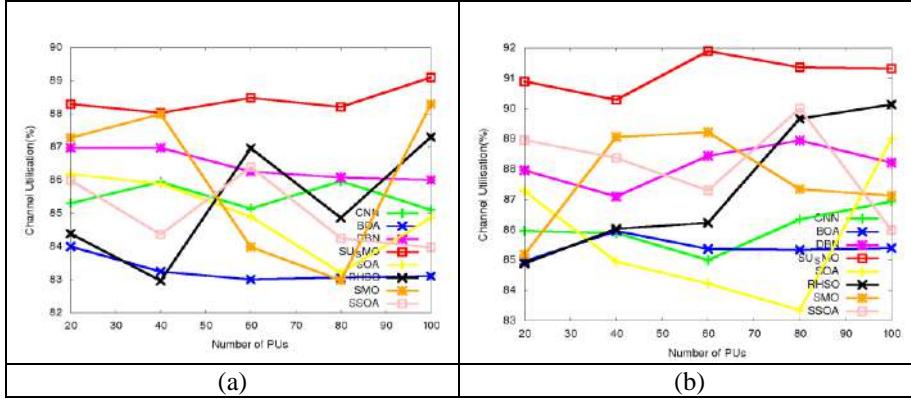


Fig:7 Evaluation on Channel Utilization of the proposed SU-SMO model in contrast to the conventional approaches by varying the PUs a) Experiment 1 b) Experiment2

5.6 Convergence analysis

Fig.8 depicts the convergence (cost) study of the suggested SU-SMO model for several iterations compared to conservative techniques like BOA, SOA, DBN, RHSO, SMO and CNN . The analysis is therefore performed by changing the iterations from 0, 5, 10, 15, 20, and 25. In reality, for greater system performance, the cost values must be at a minimal. The SU-SMO model achieves a better convergence, nevertheless, as the number of iterations rises. That is, utilising the SU-SMO strategy, a substantially lower cost value of 0.1 is seen from the 20th to the 25th iteration, which is much better than the existing BOA, SOA, DBN, RHSO, SMO and CNN approaches. The SU-SMO model's evolution is supported by the overall evaluations.

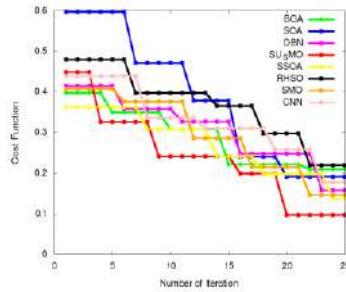


Fig:8 Convergence Analysis of suggested SU-SMO model over extant approaches

6. Conclusion

This work focuses on securing data packet delivery by foreseeing channel states (idle or busy). In this case, a SU-SMO-oriented LSTM algorithm was used to predict the channel state. Additionally, the AES technique was used to transmit

data securely in the accessible spectrum channels. Finally, analysis was carried out to show how the model could be improved. Particularly, the proposed model outperforms the following models in terms of average backoff, under 80 PUs: DCNN (0.436), BOA (0.474), DBN (0.416), SOA (0.429), RHSO (0.415), SMO based LSTM (0.371), and SSOA (0.398), while the proposed SU-SMO method achieves (~)0.375. Additionally, for 100 PUs, the RHSO and SOA reported channel utilisation of 90.67% and 98.33%, respectively, but the suggested scheme showed maximum channel usage of 91.26%. In the future, the accuracy and performance of the suggested estimator is evaluated using various cyclic prefix lengths and types.

References

- [1] Deka, S. K., Sarma, N.:Opportunity prediction at MAC-layer sensing for ad-hoc cognitive radio networks. *Journal of Network and Computer Applications*, 82, 140-151. (2017).
- [2] Cao, X., Song, Z., Yang, B.:Multi-slot reservation-based multi-channel MAC protocol for dense wireless ad-hoc networks. *IET Communications*, 12(10), 1263-1271. (2018).
- [3] Rajput, R.S., Gupta, R. & Trivedi, A.: An Adaptive Covariance Matrix Based on Combined Fully Blind Self Adapted Method for Cognitive Radio Spectrum Sensing. *Wireless Pers Commun* 114, 93–111 (2020). <https://doi.org/10.1007/s11277-020-07352-9>.
- [4] Liu, S., Han, B., Peng, W.:A polling-based traffic-aware MAC protocol for centralized full-duplex wireless networks. *IEEE Access*, 6, 28225-28238. (2018).
- [5] Mourougayane, K., Amgothu, B., Bhagat, S., Srikanth, S.:A robust multistage spectrum sensing model for cognitive radio applications. *AEU-International Journal of Electronics and Communications*, 110, 152876. (2019).
- [6] Le, L. B.:Design and optimal configuration of full-duplex MAC protocol for cognitive radio networks considering self-interference. *IEEE Access*, 3, 2715-2729. (2015).
- [7] Xu, X., Zhang, N., Song, H., Liu, A., Zhao, M., Zeng, Z.:Adaptive beaconing based MAC protocol for sensor based wearable system. *IEEE Access*, 6, 29700-29714. (2018).
- [8] Patel, J. B., Collins, S., Sirkeci-Mergen, B.:A framework to analyze decision strategies for multi-band spectrum sensing in cognitive radios. *Physical Communication*, 42, 101139.(2020).
- [9] Cheng, Y. C., Wu, E. H., Chen, G. H.:A decentralized MAC protocol for unfairness problems in coexistent heterogeneous cognitive radio networks scenarios with collision-based primary users. *IEEE Systems Journal*, 10(1), 346-357. (2016).
- [10] Guirguis, A., Karmoose, M., Habak, K., El-Nainay, M., Youssef, M.:Cooperation-based multi-hop routing protocol for cognitive radio networks. *Journal of Network and Computer Applications*, 110, 27-42. (2018).
- [11] Mansoor, N., Islam, A. M., Zareei, M., Vargas-Rosales, C.:RARE: A spectrum aware cross-layer MAC protocol for cognitive radio ad-hoc networks. *IEEE Access*, 6, 22210-22227. (2018).
- [12] Li, A., Han, G.:A fairness-based MAC protocol for 5G cognitive radio ad hoc networks. *Journal of Network and Computer Applications*, 111, 28-34. (2018).

- [13] Du, M., Zheng, M., Song, M.:An adaptive preamble sampling based MAC protocol for cognitive radio sensor networks. *IEEE sensors letters*, 2(1), 1-4. (2018).
- [14] Aloqaily, M., Salameh, H. B., Al Ridhawi, I., Batieha, K., Othman, J. B.:A multi-stage resource-constrained spectrum access mechanism for cognitive radio IoT networks: Time-spectrum block utilization. *Future Generation Computer Systems*, 110, 254-266. (2020).
- [15] Narayanan, N. S., Patnaik, M., Kamakoti, V.:ProMAC: A proactive model predictive control based MAC protocol for cognitive radio vehicular networks. *Computer Communications*, 93, 27-38. (2016).
- [16] Patnaik, M., Garg, C., Roy, A., Devanathan, V. R., Balachandran, S., Kamakoti, V.: Prowatch: a proactive cross-layer workload-aware temperature management framework for low-power chip multi-processors, *ACM J. Emerg. Technol. Comput. Syst.* (JETC) 12 (3) (2015).
- [17] Jha, S. C., Phuyal, U., Rashid, M. M., Bhargava, V. K.:Design of OMC-MAC: an opportunistic multi-channel MAC with QoS provisioning for distributed cognitive radio networks. *IEEE Transactions on Wireless Communications*, 10(10), 3414-3425. (2011).
- [18] Tobagi, F., Kleinrock, L.:Packet switching in radio channels: Part II-The hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *IEEE Transactions on communications*, 23(12), 1417-1433. (1975).

MULTI ENERGY-HARVESTING SMART WATER METER DESIGN FOR UNDERGROUND WATER PIPELINE LEAKAGE DETECTION

Hari Prakash Athinarayanan¹ and Muthupavithran Selvam²

¹ Newcastle University, Newcastle upon Tyne, United Kingdom

² De Montfort University, Leicester, United Kingdom

harish091198@gmail.com

p2519525@alumni365.dmu.ac.uk

Abstract. The water leakage detection on pipelines has improved with the advent of remote environmental- wireless sensor network (WSN) and Internet of Things (IoT) technologies over the past few years. Leakage detection in under-ground water pipelines involves algorithms to predict leakage and estimate its location, in which accuracy depends significantly on the distance between the water meters stationed at district water stations. This research proposes to deploy smart wireless water meter nodes at intermediate distances to enhance the accuracy sharply. This requires water meters to operate autonomously, harvesting energy from surroundings, to be self-sufficient. This research proposes a heterogeneous communication with smart Water Meters that focuses on transferring data and reducing energy consumption by establishing communication with wireless sensor nodes for smart water meter applications.

Keywords: Environmental monitoring · Autonomous Smart water meter · Energy harvesting · Thermoelectric power · energy storage · wireless sensor network · Internet of Things · Long Range Wide Area Network (LoRaWAN)

1 Introduction

Energy harvesting (EH) is one of the active researches in electronics, where the products aim to attain self-sufficiency by harvesting energy from ambient sources. This project's main initiative is to probe into the energy harvesting for low power operating Smart water meters to detect leakages in underground wa- ter distribution pipelines. This research focuses on the non-invasive harvesting of energy [1] with a multi-source energy harvester to scavenge energy to power the smart water meters. It uses WSN and IoT infrastructures to communicate [2], and it can be deployed at any remote location. The wireless Sensors Network technologies interconnect the individual smart water meters in the pipelines to form a network of meters that can log real-time data [3]. Generally, E-WSNs are

deployed far from inhabited centres and thus without access to electricity. To attain autonomous operation [4], the device cannot consume more power than a harvested source can provide. Otherwise, when consumption exceeds the limit of energy production, the device will eventually deplete its energy reserve and cease working. The empty energy reserve leads to undesirable system performance. Moreover, the theory of energy-neutral operation [5], and similar works have been expressed, where the main improvement is the non-ideals of energy storage devices are considered, yielding an enhanced theory. The Energy consumption was compensated in operating low power devices with EH techniques for autonomous operation. The (EH) techniques related to several sources can be implemented simultaneously to harvest more energy using multi-energy harvesters or multi-source energy harvesters [6] to meet the energy consumption demand in low-powered devices. Moreover, the proposed self-powered water meter designs [7] made the water distribution system more sophisticated and enhanced data management. Further, the possibilities of harvesting energy from different available energy sources were analysed to explore EH methods.

2 Ambient Energy Sources

2.1 Energy sources in the underground

The Underground facility of water transmission pipelines has an unlimited energy source for ambient energy harvesting, including indoor solar, piezo-electricity, Air current flow, acoustic noise, and thermo-electricity. The presence of water pipelines also allows harvesting energy from the water flow in the pipes, vibration on the pipe surface, tri-boelectric energy, temperature variations in the outer metallic surface, and acoustic noise caused by the pipes. These energy harvesting sources can be used by considering the project scenario. The preliminary analysis unfolded that a single energy source is insufficient for attaining self-sufficiency. The selection of energy sources must be considered significantly based on the application characteristics and environment. Moreover, various energy sources exist in the same environment, which supports improving the harvested energy for operating the device like smart water meters. As the energy sources coexist in nature, it is inefficient to consider only one primary source as a suitable source while designing an autonomous powered system. This negligence of not considering multiple sources limits the quantity of energy harvested from the ambient sources, resulting in a fallback of device efficiency due to insufficient power. The actual ambient energy harvesting implementation involves multiple energy source harvesting methodologies to extract as much energy from the environment to meet the load demand of the smart devices [9]. Multi-source energy harvesting is a potential solution to attain self-sufficiency, as it compensates for the fluctuations in ambient sources caused by natural phenomena.

2.2 Selection of sources for harvesting

The presence of an energy source and availability of a harvesting technology does not imply that the harvesting can be efficient. The energy harvester should be

compact, energy conversion efficient and cost-efficient to deploy for harvesting energy from the available sources. Certain factors should be considered while selecting the sources for multi-source energy harvesting [10], such as power density, the energy source's nature of deploying environment, cost-effectiveness, the efficacy of energy harvesting, and the size of the overall harvester [11]. These factors are significant while selecting sources for self-powered systems during design considerations. Considering these factors over the energy sources, the potential sources to harvest energy in underground water pipelines are water flow in pipes, thermoelectric energy, piezoelectric energy, and indoor photovoltaic energy. The energy sources such as airflow, tribo-electric energy and acoustic energy are not considered as they are inefficient and not cost-effective to implement in the project deployment environment. Moreover, the aim to harvest energy by a non-invasive approach in an underground pipeline limits the consideration of energy harvesting from the water flow inside the pipelines and the general photovoltaic harvesting for multi-source energy harvesters. Indoor solar harvesting is not efficient to implement because the presence of indoor lights in an underground pipe facility is an uncertain factor. Hence, the potential sources that can be abundant and suitable are shortlisted, which are thermoelectric and vibrational energy [12]. As a first stage approach, this project implements the Thermo-electric energy harvesting of a multi-source energy harvester in a water pipeline, considering the underground environmental factors.

3 PROPOSED ARCHITECTURE

3.1 Network Architecture Design

The core idea of this research is to deploy sensor nodes at intermittent distances, which are connected by ZigBee and LoRaWAN wireless communication networks with the gateway device, as shown in Fig. 1. The architecture diagram provides a design consideration to interconnect the smart water meters to IoT web server for monitoring and detecting leakages more efficiently. The proposed architecture consists of three segments: End Nodes, Gateway, and Network server. The end nodes are the clusters formed by interconnecting sensor nodes (smart water meter), in which each cluster has several sensor nodes and a cluster head node.

The sensor nodes of each cluster are powered by energy harvested from multiple ambient energy sources to make the sensor nodes work in autonomous mode. These sources' energy density is dependent on multiple uncertain factors. Hence, the cluster nodes are powered from an energy storage device attached to the harvester unit of the smart water meter. The Cluster heads transmit the collected data to the gateway device, which further sends the data to the IoT web server. The leakage prediction and estimation algorithms are performed over the data stored in the server.

3.2 Energy Harvester Architecture Design

The environmental factors favour harvesting energy in the outdoor environment while providing limited support in the underground environment. So, suitable

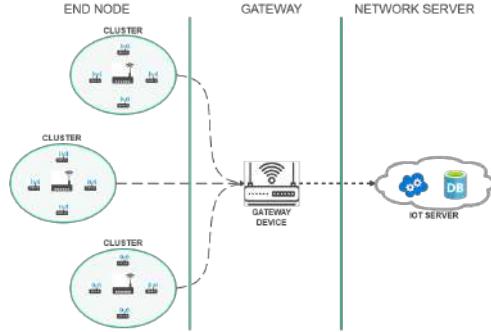


Fig. 1. Network architecture of smart water meter.

and profitable energy sources must be harvested by using the multi-source energy harvester to support the smart water meters. The identified sources include thermoelectric energy sources, vibration energy sources and indoor light sources. These sources contribute to the operation of the autonomous smart water meter, where the thermoelectric source dominates as a significant energy contributor. The thermoelectric energy is predominant as it is abundant and can be harvested more by increasing the number of transducers, whereas the indoor light dependent PV cell based energy transducer is limited by the availability of lighting device and its brightness, and the vibrational energy harvesting is highly limited as the pipelines are fixed stably by mechanical structures for stability. The Fig. 2 shows the topology design for the multi source energy harvester in terms of block diagram.

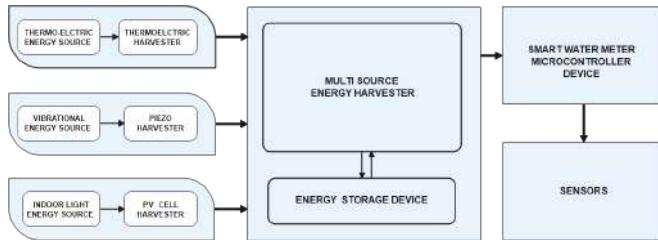


Fig. 2. Multi-source energy harvester design topology.

Thermometric converters can be realized as alternating current sources combined with an energy-storing capacitor. The Thermo-electric energy is harvested with thermometric harvester (TEH) cells attached to the outer surface of the pipeline. The current generated is solely dependent on the temperature changes

in the harvester material, as related by the following equation..

$$I(t) = a_0 A \frac{dT}{dt} \quad (1)$$

Where $I(t)$ is the current generated by the TEH device at the time instance t , a_0 is the thermoelectric material coefficient vector, A is the surface area of the material plates and dT/dt represents the change of temperature over time. The TEH cells are connected in series and parallel combinations to attain the necessary voltage threshold for the energy storage device to store the harvested energy. The MEMS piezo vibration energy harvester converts the micro-vibrations into valuable electrical energy, which further can be conditioned by the harvester circuit. The Indoor light energy harvester is a special consideration, as it is only used where the indoor lights are available.

The energy harvested from ambient energy sources has different energy densities. Hence, the energy harvested is stored in an energy storage device such as a large capacitor. The energy storage capacitor accumulates the harvested energy and provides the stored energy to the Micro-controller in the burst mode.

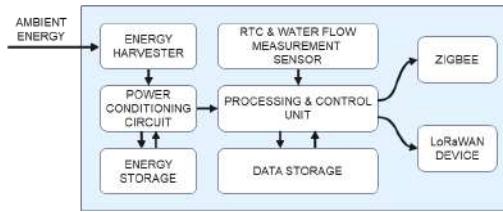


Fig. 3. Smart water meter cluster head node-Single perspective block diagram

The smart water meter master nodes have several internal components in the circuitry, as shown in Fig. 3, to harvest ambient energy to meet the power demand for data transmission. The energy estimation methodologies help in calculating the power requirements to operate the node in burst mode, which helps in selecting storage device, Micro-controller and in implementing energy-efficient methodologies to save power during the operational time of the device. The Specific research challenges are concentrated in the energy transfer process from multiple energy harvesting sources to the energy storage device.

3.3 Heterogeneous Data Propagation Wireless Network Model

The wireless network formed by the smart water meter nodes with a homogeneous network structure has less efficacy in data transmission. The Fig. 4 represents a heterogeneous communication architecture of the design, where the devices are connected with a Micro-controller through a short-range protocol such as IEEE 802.15.4. Further, the Micro-controller communicates with devices

at a distance of kilometers by employing the LoRaWAN protocol, specialized for low power systems to decrease the power consumption of the Water Meter. LoRa protocol is a fitting technology that achieves our work's purpose to deploy low-power performance for long-range communication.

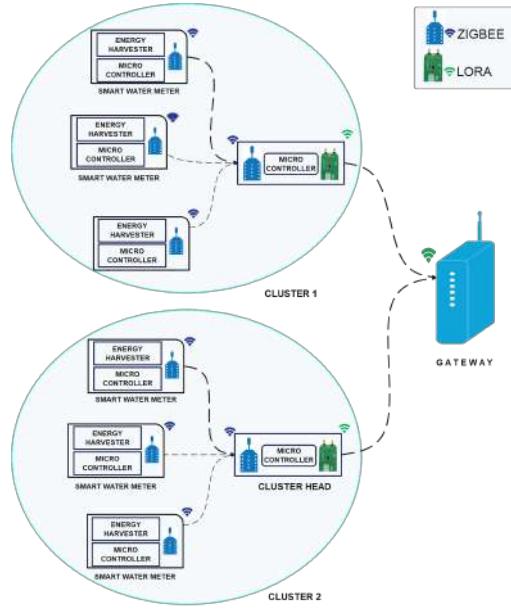


Fig. 4. Network Architecture of Smart water meter system

Short-Range Data Communication Network The short-range communication is achieved by a Micro-controller node with an XBee-S2 module configured as End devices and a Coordinator device, and thus forming wireless sensor nodes and a cluster head. The XBee-S2 wireless device supports point-to-point communication to exchange data within 120 meters. Therefore, the nodes are fixed within the range of 100 m for a better operation of data transmission. The nodes within the cluster range will send data to the nearby cluster head, in which cluster nodes and cluster head are assigned and identified by a device address.

Long-Range Data Communication Network A heterogeneous communication structure with a long-range protocol, such as the LoRa, presents a trade-off between energy consumption, signal strength, and latency. The network structure has several clusters located at around a few kilometres between them. The cluster nodes transmit data to the corresponding head node, which operates as a secondary gateway device in the network. The range of the long range communication of the wireless sensor nodes are between 100 m to 5000 m.

Long-Short Data Communication Network The long-short network protocol is an efficient combination of Zigbee and LoRaWAN based on the requirement of the deployment scenario. The short-range protocol (Zigbee) is implemented only within the clusters formed by the smart water meter nodes, and the long-range protocol (LoRaWAN) is used to transmit telemetries data to a gateway device which is located at a far distance from clusters and is considered as a shared resource in the network architecture as shown in Fig. 5. Each wireless node acknowledges the presence of itself by intimating its presence to the cluster head at fixed intervals. The data transfer frequency depends on the availability of the energy in the Energy storage device. When the sufficient energy is harvested, the microcontroller is woken up from sleep mode and will start initiating data transfer process. The microcontroller initiates a data read and data transfer processes, when the energy in the storage is sufficient for performing a data communication. The cluster head collects the data from its cluster nodes and transmit the data to the gateway using LoRaWAN protocol. The gateway device located at the district water station receives the data from the cluster heads of different clusters.

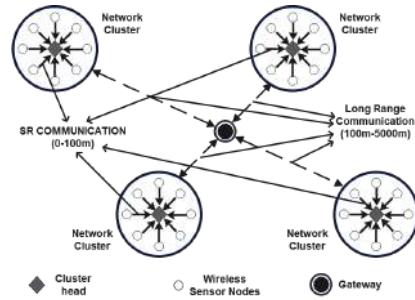


Fig. 5. LoRaWAN Network Of Smart Water Meters

3.4 Web Server

The telemetric data from different nodes located at intermittent distances between the district water stations are sent to the IoT web server using HTTP or MQTT protocol from gateway device to process and predict water leakage using the algorithms. Generally, the Network server has a database where the telemetric data are stored. The method of deploying nodes in-between may increase the data stack, but these data are significant for improving leakage prediction accuracy as the analysis is performed over each virtually segmented water pipeline. The corresponding API can be used to access these data. The rule engine allows the analysis of telemetric data and provides the estimation results.

4 ELECTRIC ENERGY HARVESTING FROM WATER PIPELINES

The thermoelectric energy harvesting method [13], [14] mainly depends on the thermoelectric generator (TEG), which acts as a transducer device for converting thermal energy into electrical energy. The thermoelectric energy harvesting technology is formulated based on the Seebeck effect. This effect describes the conversion process of temperature gradient into electric power that occurs at the junctions of the thermoelectric elements of a thermoelectric harvester (TEH) device. TEH device is a portable and reliable energy converter that can generate electrical energy in applications where the heat is dissipated unprofitably. The TEH as an energy harvesting module is used in many potential applications such as medical devices, wireless devices and consumer electronics.

4.1 TEG Power Harvesting System

The thermoelectric energy harvesting system [15] is a collection of subunits interconnected procedurally and tuned to extract energy efficiently from the source. It consists of four major subsystems, as shown in Fig. 6, comprising heat sources, thermal harvesters, Dc/Dc converter, control unit, and an energy storage unit. The project uses the TEH cells having a dimension of $30 \times 30 \times 3.4$ cm with a

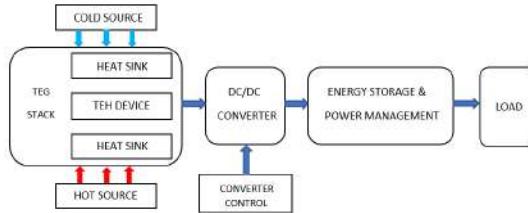


Fig. 6. Block diagram of thermoelectric energy harvesting system with a TEH stacks

semi-flexible material structure to harvest energy from the pipeline. Eight TEG modules are connected to form a TEG harvester grid (as shown in Fig. 7). This arrangement in the grid can be altered anytime to match the load demand. Connecting the TEG devices in series gives the higher scaled output voltage. The scaling is based on the number of devices connected to the grid. The TEG device connected in a parallel connection produces a higher current at a fixed voltage, thus providing sufficient power to drive the load. The TEG produces unregulated output power, and so it must be regulated to use it directly or to store it in energy storage devices. To improve both the parameters, namely voltage and current, to match the load input and operating requirement, the TEG devices can be connected appropriately in series and parallel to form a hybrid TEG grid. Here,

eight TEH cells are connected together, forming two parallel circuit lines, with each line consisting of four TEH cells in series. The DC/DC converters regulate

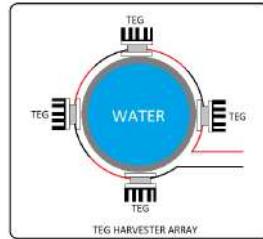


Fig. 7. Cross-section view of TEG harvester grid positioning over pipeline

the output DC power by stepping down or stepping up the voltage to match the output load requirements. Since the output voltage of the TEG is lower than the load requirements and not constant, the DC/DC converter becomes significant in the TEG harvester system. The DC/DC boost converter requires a controlling unit to control the transistors' firing period and thereby control the duty cycle of the converter component (MOSFET). Hence, to obtain maximum power from the available input energy from the TEG device, an algorithm has to be implemented to keep track of instantaneous power and fine-tune the converter's duty ratio. The Maximum power Point Tracking (MPPT) algorithm is implemented to tune the duty cycle on the operating time dynamically. The harvested energy is stored in the energy storage device, either a battery type or a storage capacitor type device that can store energy to be retrieved to power the load. The stored energy can then be used directly to power the devices. This stored energy can be released instantaneously from the Capacitor with low internal resistance, or released slowly with energy buffer circuits.

The output of the Dc/Dc converter is adjusted to match the input requirements of the storage element. For example, the input of a 5v capacitor must be allowed utmost till 4.8v to protect the Capacitor from the device's failure caused by overcharging. It is acceptable to supply 5v input as the storage devices have its tolerance limit. So, the device's output should be lower than or equal to 5v to charge the 5v storage device. The energy storage subsystem is a crucial component of a sensor node, significantly affecting its overall efficiency. The choice of energy storage technology also affects a smart device's size, cost, and operating life [16].

4.2 Harvested Energy Management

The power management circuit is significant in managing the power stored in the energy storage device by efficiently regulating the input and output power and protecting the primary circuit. In this prototype, the energy obtained from

the harvester grid (TEG) is stored in the energy storage capacitor. The energy harvested from the sources is fluctuating based on the water temperature. The Capacitor charges to its maximum voltage unless restricted by an external circuit. Charging the Capacitor to its fully charged state may cause permanent damage to the Capacitor due to overcharge. Hence, the energy storage device requires a power management device and circuits to manage the power. It allows the DC/DC converter to regulate the input power by adjusting its duty ratio by implementing the MPPT algorithm to scavenge maximum available energy. It also protects the energy storage device from Over-voltage and under-voltage to maintain its health. The power management circuit uses the BQ25570 ic, an ultralow-power Power management Integrated Controller (PMIC) for power management, which has an in-built DC/DC boost converter controlled by an MPPT controller unit a cold start circuit and a buck converter.

4.3 Experimental Design And Setup

The experimental prototype consists of energy harvesters, power management hardware, a storage device, and a load. The hardware devices used in the project prototype include THE device (8 cells) as ambient thermoelectric energy harvesters, BQ25570EVM-206 as power management hardware, and Capacitor ($5v, 100\mu F$) as a storage device and MSP430fr5994 Ic circuit with water flow sensor together as load. The power management circuit with BQ25570 manages the power obtained from the TEG harvesters. The harvested energy is stored within the capacitor storage device, and the MSP430fr5994 device with wireless communication device acts as a load device. The input voltage rating of the Msp430fr5994 device is 1.8v to 3.6v. Hence, it is sufficient to provide the voltage directly from the output terminal without activating the buck converter in the BQ25570 IC. The test setup was made by attaching the TEG module and the heat sinks to the water pipeline. The harvester's output is measured using a digital oscilloscope connected to the output terminal of the BQ25570 hardware circuit.

Steps to assemble the hardware are:

1. The TEG devices should be connected in series electrically, and the output terminals are connected to V_{in} and GND terminals of the $J1$ connector.
2. The Capacitor is connected to V_{Bat} and GND terminals at the $J8$ connector.
3. At $JP2$ set of pins, enable pin should be connected to the Ground pin using wire or a shunt.
4. At $JP3$ sets of pins, the V_{outEN} pin is connected to the GND pin using a wire or a shunt.
5. Fix the shunt between the $V_{oc-sample}$ and 50 percent in the $JP4$ pins.
6. Disconnect the shunt between the $JP1$ set of pins.
7. The output is obtained from the $JP11$ connector.

In step 3, the Enable pin is connected to the ground to enable the IC. The V_{out-EN} pin is connected to ground at step 4 to disable the buck converter and get output directly from the Capacitor, bypassing the buck converter. In step 5, the $V_{oc-sample}$ is connected to a 50 percent pin because the maximum power point lies around 50 percent of the open-circuit voltage for the TEG device.

Buck Converter 'On' Mode The Buck converter from the BQ25570 chip can be enabled by connecting the V_{out-En} with the $bat-ok$ pin in the $JP6$ pins. The output of the buck converter is the scaled-down value of the capacitor voltage. The regulated buck converter output is helpful for the operating systems at a very low voltage like $1.5v$. A high-efficiency PFM based controller controls the buck converter. The output of the buck converter is always maintained at $1.8v$ at the output terminals.

Buck Converter 'Off' Mode The results of the steps in turning Off the buck converter and connecting the output ports directly to the capacitor storage. In this mode of operation, the Buck converter is switched off to get a higher output voltage. At this mode of operation, the TEG modules are connected in series and fit the water pipeline, and the output of the harvester is measured using a digital oscilloscope device. Further, the output of the BQ25570 circuit is $3.17V$, which can be directly fed to the Msp430fr5994 ic as its input requirement is met at this voltage level.

4.4 Simulation

The energy harvester model for thermoelectric harvesting was developed in Matlab simulation software to observe the system at various temperatures and with increased TEG devices. This harvester model also includes the process within the PMIC hardware. It consists of subsystems such as the TEG model, Overvoltage and under-voltage protection models, Boosts converter model and capacitor storage model. To estimate the coefficients of the harvester, the number of cells in series is set to 1, and the hot and cold temperatures are fixed to $49.7^{\circ}C$ and $33.2^{\circ}C$, respectively (given by the manufacturer in the datasheet for a standard reference value) and simulated. A similar circuit consisting of the same configuration but with four cells is connected in parallel and four cells in series with the simulation circuit. The model and the simulation waveforms are obtained from simulating the setup. The output of the TEG model is connected to the Overvoltage protection model of Capacitor. Further, the output is given to the DC/DC boost converter circuit controlled by an MPPT controller model [17], providing the duty ratio for the converter. Further, the boost converter's output is fed to capacitor storage as input. Then, there is an under-voltage protection model to monitor the voltage level of the Capacitor and a trigger circuit that triggers a switching device that connects the load to the Capacitor when its voltage level reaches a value of $3.2v$. In the case of overvoltage situations or under the absence of load, a separate parallel circuit with a shunt resistor is connected to the TEH array.

5 RESULT

The geometrical shape and structure of the heat sink also greatly influence maintaining a thermal gradient in the TEG device. The proposed system that insulates the inner side heat sink by a thermal insulator will help maintain the

temperature at that side without the interference of surrounding air. This will improve the output of the TEG device significantly. The TEG stack formed by sandwiching the TEG device between heatsinks and fitting to the water pipeline by connecting TEG devices in series to scale up the terminal voltage will help the Capacitor charge quickly, thereby providing power to the load device. The output produced by a single TEG device might not have been more helpful, but the array of TEG devices drastically improves the energy harvested and thus making it a potential harvesting energy source.

The PMIC hardware circuit (BQ25570) supports burst power mode powered by a storage capacitor. The energy harvester built with TEG devices, BQ25570 based Power management circuit and Capacitor, harvests energy with the usage of MPPT controller for boost controller operations to meet the load demand of smart wireless device loads. The energy harvested from the TEG array containing TEG stacks connected in series is continuously fed to the storage capacitor via boost convertor and is further supplied to the load device. The boost converter duty ratio is tuned by the MPPT controller circuit every 16 seconds. The MPPT controller circuit interrupts the circuit every 16 seconds and updates the maximum power point by storing the corresponding voltage to the reference voltage capacitor, which is used as a reference for adjusting the duty ratio of the boost converter. The output buck converter is turned off, as it does not require in the project scenario. The data obtained from the practical experimentation is simulated in the simulation model to validate the simulation results. The simulation model used the same process flow from the hardware to design the equivalent thermoelectric harvesting model. The simulation model could produce an output as nearly as the physical experiment readings. The Hardware

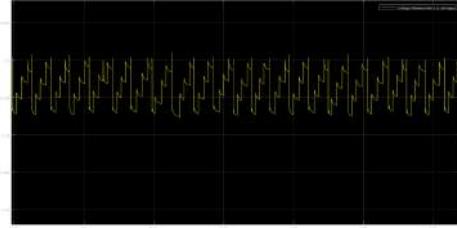


Fig. 8. Waveform Showing Charging And Discharging Of The Energy Storage Capacitor

setup is fitted over the outer surface of the water pipeline to log the data for experimentation. The energy storage device accumulates energy obtained from the transducer belt containing TEH cells, which is conditioned by the PMIC circuit. The PMIC charges the energy storage device until the preset voltage threshold of 3.2v. Once the sufficient energy is available, the main microcontroller reads the data from the sensors and transfer the processed data to the cluster head. The Fig. 11 shows the waveform displaying the voltage across the

Capacitor while charging and discharging occur in the PMIC hardware circuit. The voltage gets built in capacitor in steps up to $3.2v$, then the PMIC hardware circuit connects the Microcontroller of smart water meter to the energy storage capacitor, causing a drastic drop in voltage across the Capacitor.

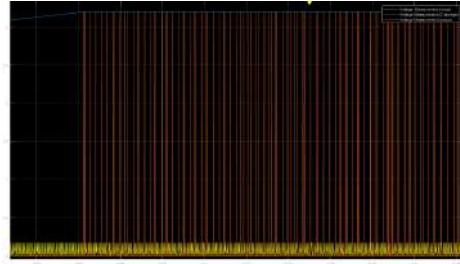


Fig. 9. Output Graph Of A Simulation Showing Input, Output And Capacitor Voltages

The simulation model waveforms shown in Fig. 12 can also be observed to infer that the output generated is nearly the same as the physical readings obtained by the experiment. The simulated harvester produces a fixed output voltage level of $3.2v$ (ideal case) in its output, and the real-time measurement obtained is $3.17v$ when the capacitor value is above the threshold voltage of $3.2v$, which can be observed from Fig. 13. Thus, the harvester prototype helps attain self-sufficiency in terms of power by adding different energy source harvesters and the present harvester or by scaling up the number of TEG devices.

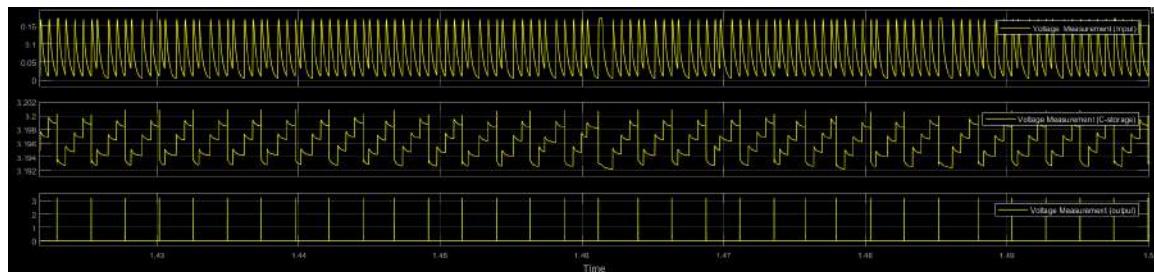


Fig. 10. Output Waveforms Of The Simulation

6 CONCLUSION

The research aims to deal with the challenges of harvesting energy from underground water pipelines without a non-invasive approach to the pipelines. The

deploying location of the system restricts access to primary harvest sources of energy. So, the project aims to develop a multi-source energy harvester to overcome unfavourable environmental factors. The multi-source energy harvesting is a suitable solution to compensate for the absence of quality energy sources like light sources. It tries to compensate for quality by utilizing quantity. The project implements the thermoelectric harvesting process in the water pipelines, considering the underground scenario with a non-invasive approach.

The TEG devices were studied to harvest energy from the thermal gradient by fixing them to the pipes. Every minor increase in the output power inside the underground facility is advantageous for the system to operate with high efficacy. After considering the effects of thermal buffers, all the TEG modules were equipped with heat sinks. Now the TEG modules are connected to form an array in which the TEG configurations were investigated, and series mode alone is implemented in this project due to the unavailability of more TEG devices. As investigated, the TEG modules can be interconnected in series or parallel or hybrid ways based on load requirements. These configurations also influence the charging action of the Capacitor. The BQ25570 power management circuit is used to harvest energy efficiently, manage the power effectively, and protect the storage element. This hardware containing an MPPT controller automatically implements the MPPT algorithm to improve the output harvested energy.

The practical experiments were conducted to study the working of this power management device. The device was made to operate by switching off its buck converter. The test results were recorded separately. The simulation was performed by creating an equivalent model of the physical system. The output waveforms validate the model's correctness by providing nearly the same output as the recorded data in practical experiment, which further validates the research approach. The project achieved its aims and objectives thoroughly with the prototype. It was able to cross some of the stepping stones to create an autonomously powered system.

7 FUTURE WORKS

The project dealt with a single power source for the initial approach stage, but could adapt to any energy source by slight modification in its harvester unit. The challenges in harvesting other ambient energy sources will be studied, and their harvesters will be added to the circuitry to develop a full-fledged self-powered system in underground water pipelines. On combining various energy harvesters, voltage imbalance can occur in the circuit due to uneven energy generation. This will be studied and solved before implementation in the underground water pipelines. The wireless network architecture proposed in this paper will be implemented further to upload and analyse the telemetric data in the IoT cloud server.

References

1. Sharma, R.R. (2021) "Gas leakage detection in pipeline by SVM classifier with automatic eddy current based Defect Recognition Method", Journal of Ubiquitous Computing and Communication Technologies (UCCT), vol. 3, no. 3, pp. 196–212, September 2021.
2. F. Shaikh and S. Zeadally, "Energy harvesting in wireless sensor networks: A comprehensive review", Renewable and Sustainable Energy Reviews, vol. 55, pp. 1041-1054, 2016.
3. I. Stoianov, L. Nachman, S. Madden and T. Tokmouline, "PIPENET a wireless sensor network for pipeline monitoring", Proceedings of the 6th international conference on Information processing in sensor networks - IPSN '07, 2007.
4. O. Ozel, K. Tutuncuoglu, S. Ulukus and A. Yener, "Fundamental limits of energy harvesting communications", IEEE Communications Magazine, vol. 53, no. 4, pp. 126-132, 2015.
5. A. Kansal, D. Potter and M. Srivastava, "Performance aware tasking for environmentally powered sensor networks", ACM SIGMETRICS Performance Evaluation Review, vol. 32, no. 1, pp. 223-234, 2004.
6. J. Colomer-Farrarons, P. Miribel-Catala, A. Saiz-Vela and J. Samitier, "A Multi-harvested Self-Powered System in a Low-Voltage Low-Power Technology", IEEE Transactions on Industrial Electronics, vol. 58, no. 9, pp. 4250-4263, 2011.
7. R. Garg and N. Garg, "Energy Management in a Multi-Source Energy Harvesting IoT System", Journal of Information Technology Research, vol. 13, no. 2, pp. 42-59, 2020.
8. O. Akan, O. Cetinkaya, C. Koca and M. Ozger, "Internet of Hybrid Energy Harvesting Things", IEEE Internet of Things Journal, vol. 5, no. 2, pp. 736-746, 2018.
9. G. Tuna and V. Gungor, "Energy harvesting and battery technologies for powering wireless sensor networks", Industrial Wireless Sensor Networks, pp. 25-38, 2016.
10. M. Prauzek, J. Konecny, M. Borova, K. Janosova, J. Hlavica and P. Musilek, "Energy Harvesting Sources, Storage Devices and System Topologies for Environmental Wireless Sensor Networks: A Review", Sensors, vol. 18, no. 8, p. 2446, 2018.
11. A. Randriantsoa, D. Fakra, L. Rakotondrajaona and W. Van Der Merwe Steyn, "Recent Advances in Hybrid Energy Harvesting Technologies Using Roadway Pavements: A Review of the Technical Possibility of Using Piezo-thermoelectrical Combinations", International Journal of Pavement Research and Technology, 2022.
12. N. Shah and S. Sundar, "Smart Electric Meter Using LoRA Protocols and IoT applications", Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018.
13. D. Enescu, "Thermoelectric Energy Harvesting: Basic Principles and Applications", Green Energy Advances, 2019.
14. F. Akhtar and M. Rehmani, "Energy replenishment using renewable and traditional energy resources for sustainable wireless sensor networks: A review", Renewable and Sustainable Energy Reviews, vol. 45, pp. 769-784, 2015.
15. S. Kim and P. Chou, "Energy Harvesting with Supercapacitor-Based Energy Storage", Smart Sensors and Systems, pp. 215-241, 2015.
16. T. Yajima, K. Tanaka and K. Yazawa, "Thermoelectric On-Spot Energy Harvesting for Diagnostics of Water Service Pipelines", 17th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITHERM), 2018.
17. H. MAMUR and Y. COBAN, "Detailed modeling of a thermoelectric generator for maximum power point tracking", Turkish journal of Electrical Science & Computer science, vol. 28, no. 1, pp. 124-139, 2020.

Tensor Flow Model with Hybrid Optimization Algorithm for solving Vehicle Routing Problem

Jai Keerthy Chowlur revanna¹ and Nushwan Yousif B. Al-Nakash^{†2}, PhD
Harrisburg University of Science and Technology PA 17101, USA
326 Market St, Harrisburg, PA 17101, USA
jchowlur@my.harrisburgu.edu, nal-nakash@harrisburgu.edu

Abstract. Vehicle routing and path management system improves the best key point of selecting the path of the vehicle to move. The applications that are used for delivering the products utilize Google data to organize the vehicle movement and its coordinate positions. The traffic level indicator and the speed of vehicle movement validate the Vehicle Routing Problem (VRP)- route. Since there is another important parameter that needs to consider for the delivery process. In that, the application needs to validate the amount of traveling time and the length through which the vehicle can travel to deliver the products. This requires a better prediction model to estimate the multiple parameters of vehicle routing problems. In the proposed study, a Tensor Flow-based routing path prediction approach was chosen to train and predict the best route using the attribute weight matrix. From the parameters of the distance between the coordinates with the amount of traffic range and other related attributes, the Tensor-Flow model forms the rule to train the machine for predicting the route for vehicle movement. This model updates the learning model based on the change in parameter value and its range. The experimental result compares the suggested work to the current model of optimum VRP prediction approaches.

Keywords: Vehicle Routing Problem(VRP), Ant Colony Optimization(ACO), Tensor Flow Model(TF), Optimal Routing Path and Multi-Objective Learning.

1 Introduction

The Vehicle path identification and prediction process in area of coordinate position and grouping represents a dynamic update of data from the database to analyze the state of routing path prediction in coordinates and mining system for VRP dataset. This will be updated for a period interval that is processed in frequent order. There are several methods in classification and analyze the database contents like neural techniques and other machine learning technics. In that, sequence pattern method with the several Neural Network were most used to classify and match the relevant features from database [1]. Since, for the huge amount of node characteristics in data, it struggles in predicting class with proper attributes of feature vector. This searching process predicts the relevant feature that compares the input query with the database in high-speed analysis of large path data. In the recent research work, the matching prediction is complicated due to more irrelevant features present in the database [2]. To improve

the performance of feature identification in the searching process, machine learning technique helps to predict the best matching between the query data and feature sets in database. There are several type of machine learning techniques like Support Vector Machine(SVM), Relevant Vector Machine(SVM) and other methods. Since, in the recent days, the Neural Network and Deep Learning technique takes place a major role of data analysis and match prediction among the bulk amount of raw data.

For this condition, it directs into the Routing based data monitoring and controlling system by remote locations [3]. This is to provide data management and optimized travel to database of hospital management. In the data travel process, it needs to travel through the worldwide connectivity [4]. In that, there are lot of data patterns that can capture and identify the data which can change the database information about the user path parameters and about medicinal value that are ordered by people. This needs to be prevented by highly optimized data travel and storage system in the cloud environment [5]. For data management and predicting there are lot of prediction and re-routing techniques available to optimize the data that can protect data who doesn't have the encryption technique. Still, there are some limitations that are in the data storage and even to retrieve the database.

To solve this management and predicting problem in Routing environment, this work proposes a novel Routing management and predicting system integrated with Tensor Flow (TF) model. In this model, the TF manage the path that vehicle can travel by identifying the properties / features of data from routing path traffic level to predict the normal / range of traffic pattern by using the Ant Colony Optimization with Genetic Algorithm (ACO-GA). If this was identified as normal, then the normal flow of data travel will take care. If this was identified as pattern type of the traffic range, then this was reported to controlling system or to the routing systems to predict the data pattern at the stage of initial flow. Then this was also forward to the training model of TF to update the characteristics of the data pattern and arrange the features of it. This will enhance the feature learning of TF and also the characteristics and parameters of data pattern up to time instant. This can be achieved by the Spatial Pattern Super Learning (SPSL) method. This analyses the parameters by probabilistic distributional features to update the learning model. This can identify the multiple combination of parameters to group it and form as the cluster for better prediction process.

The primary objectives of this paper are

1. By carefully choosing the best route for data forecasting, to improve the prediction performance of routing.
2. To increase the speed of vehicle movement and the reliability data processing on vehicle, optimization method using ACO-GA was designed.
3. To estimate the location of users based on the routing properties for improved data analysis and forecasting
4. Tensor-Flow is introduced into the process of generating parameters to facilitate the analysis of the multi-objective parameters of the optimization model.

The rest of the other sub-sections are organized and elaborated as mentioned below. Section 2 gave a study and review of the current VRP model, along with its benefits and drawbacks, based on the proposed enhancements. Section 3 explains about the proposed optimal routing path selection system based on the ACO-GA with Tensor-Flow Model. Section 4 evaluates the effectiveness of suggested VRP model with the prediction rate, and it contrasts the estimated findings with the conventional methods to demonstrate the superiority of this new implementation. In Section 5, which concludes the paper and summarizes the future recommendations on this work.

2 Related works

Here a critical review is performed on different optimization models and the routing algorithms that are to optimize the vehicle routing problems. In that, the merits and the demerits of each model were explained and validated.

The benefit of this plan was that it guaranteed the vehicle's reliability and routing. An adaptive technique for effectively boosting the accuracy of safety message forecasting on the vehicle was introduced in [6]. Here, the error recovery probability rate of data forecasting was calculated using the Adaptive Byte Hybrid Automatic Repeat Request (AB-HARQ) approach [13]. A dynamic virtual bat algorithm [13] In the development of a method for supplying routing to VRP with a reduced latency frequency [7]. The objective here is to combine the benefits of Simulated Annealing (SA) and Particle Swarm Optimization (PSO) to improve the performance of optimization during path selection. A new data forwarding strategy was suggested in [8] for raising the VRP's overall performance rate. Here, traffic information were used to guarantee accurate vehicle forecasts. A sensor clustering strategy was developed in [9] to address the vehicle's concealed terminal problem, reliability, and resource scarcity problems. The applications of privacy preservation, target tracking, and misbehavior detection may be appropriate for this clustering technique.

A summary of data dissemination techniques and the significance of QoS for VRP are given in [10]. The difficulties with connection stability and energy consumption have been examined in this paper, along with the best methods for enhancing VRP performance. A unique algorithm for decreasing the broad-cast storms on VRP was devised in [11]. This work's strength was that it established the emergency message forecasting in the simplest possible way. A new V2VR approach for ensuring VRP routing was introduced in [12]. In this case, a routing choice system based on the Manhattan mobility model was used to shorten the longer predicted distance. In this paper [13], a data distribution strategy for enhancing the efficiency of data delivery via VRP is suggested. To distribute the data among the users, a probabilistic forwarding mechanism was used[23]. This work's benefits were less message latency

In [14] evolved a Batch Verification Certificateless Ring Signature (BV-CLES) technique [21] for making sure reliable navigation and dependable information forecasting on VRP. The primary objective of this research was to effectively reduce the computational overhead and delay of the vehicle for VRP connection. In addition, the transportation of the vehicle was accelerated by implementing a signature verification

method. The most important contribution of this research was that it offered a faster routing system for automobiles and lowered computing expenses. A decentralized parameters control system was suggested in [15] to enhance VRP's information forecasting. This work used a lightweight mutually relevant prediction approach to raise the vehicle's routing level. The following is a list of the main factors that this work took into account: licensed parameter storage, licensed parameter management, mutual relevant prediction, updated parameters, and revocation. Additionally, distinct categories of missing data, such as those caused by collusion, DoS, and resisting internal missing data, were identified in this work based on the routing analysis. The need to lower computational overhead, computational cost, and storage overhead still exists..

For improving the routing of VRP, [16] used a hybrid conditional privacy preservation approach. This work's main goal was to use a privacy-relevant prediction algorithm to address the identity revocation problem and lessen computational overhead. In this instance, this anonymous identity was considered the local short-term identifier accountable for signing the safety-related communication. In addition, a bilinear pairing based on the cyclic groups of the bilinear map was performed [21]. This technique's design objectives included efficiency, pertinent prediction, confidentiality, revocation, and privacy. This mechanism's benefits included improved speed and robustness with effective message-relevant prediction. A reliable communication over VRP can be established by using an improved routing method that uses the Multipoint Relay (MPR) scheme, as described in [17]. The goal of utilizing the OLSR algorithm in this case was to enhance the MPR selection strategy in order to prevent data reforecasting. Additionally, a number of metrics including delivery ratio, throughput, and vehicle delay were estimated

A self-checking procedure was used in [18] to increase the routing VRP data transfer. This system was divided into four stages: registration, relevant prediction, missing data detection, and self-checking [24]. By assuring randomization, this approach primarily aims to safeguard the vehicle from missing data users. For the purpose of ensuring the routing and confidentiality of VRP, [19] adopted a Comprehensive Identity Relevant Prediction Scheme (CIAS). In this study, asymmetric processing and pertinent prediction procedures were used to build trust amongst the entities [25]. Finding a damaging lacking data against the automobile and offering appropriate, pertinent prediction solutions were the main focuses of this work. Block chain and traffic flow computation procedures were used to build an effective routing architecture for VRP in [20][23]. This architecture includes the service the road traffic computing layer, service layer and the perception layer [13]. In which the perception layer was used to enhance the routing of the vehicle during data transfer. The drawback of this architecture was that it necessitated increasing the vehicle's overall routing and data forecasting pace.

Here prediction models are used based on the Confusion matrix. The primary classifiers are F1 Score, precision, accuracy, and recall which are sued as prediction model in the paper [27]. The confusion matrix is necessary to determine the classification accuracy of the machine learning algorithm while categorizing input into their respective labels. Below are the definition of each terms:

Accuracy: It is the ratio of occurrences of correctly categorized data to the total number of instances of data.

Precision: A strong classifier must preferably have a precision value of 1 (high). When the numerator and denominator are identical does precision equal 1.

Recall: Recall must preferably be equal 1 (high) for a classifier to be effective. When the numerator and denominator are equal does recall equal 1.

F1 score: The F1 score is only good if both accuracy and recall are indeed high. F1 score is the arithmetic average of recall and precision and is a more accurate measurement than precision.

3 Methodology

This section discussed optimal vehicle routing models with the optimization functions and the other related parameter-based categorizations. In this, the overall process was segmented as the following stages:

- Preprocessing
- Optimization
- Tensor Flow Prediction

Fig. 1 shows the architecture of ACO used with Tensor flow which is the proposed work in this paper. In this, the pre-processing covers the initial cluster of data to organize the coordinate position of users and the initial weight estimation. Then from that, the optimization functions retrieve the best routing path and the minimum distance-based path selection. Since, the optimization models are tuned to select the best routing path for vehicles based on their objective function design and the weight calculation of each path attributes. The capacity and demand are decided based on the customer needs in the area. The capacity is estimated based on Solomon benchmark data set for each case [26]. For example, in Case C101 the capacity is estimated based on the number of customers. In this paper the capacity is estimated for 100 customers and the demand is served according to the customer's needs. Similarly, for R101, RC101, R103 and all the cases shown below in the figures are defined. The demand is the number of locations requested to be delivered. The demand is the number of locations requested to be delivered. Once the capacity estimation is performed, demand is served.

The following sub-sections explain the detail structure of routing path selection. The parameters are defined by time interval, no. of iterations, No. of customers. All the above parameters are based by counting the number of vehicles and locations. The parameters are initialized as shown by algorithm 1 below from p to 1, that starts the loop from 1 and by repeating the iterations it selects the best path. As shown in table 1 different parameters were also compared with ACO GA and it is shown by the tensor flow output in table 2 that ACO GA has increased the performance compared

to other methods. Regarding crossover and mutation for the GA algorithm, Crossover is the primary operation, mutation is the secondary. As mentioned above crossover is primary which is 90% and mutation is 10% . Mutation alters the value of parameters here just like the genes and crossover is a special process that is employed to alter the sequencing of chromosomes through one generation to the next.

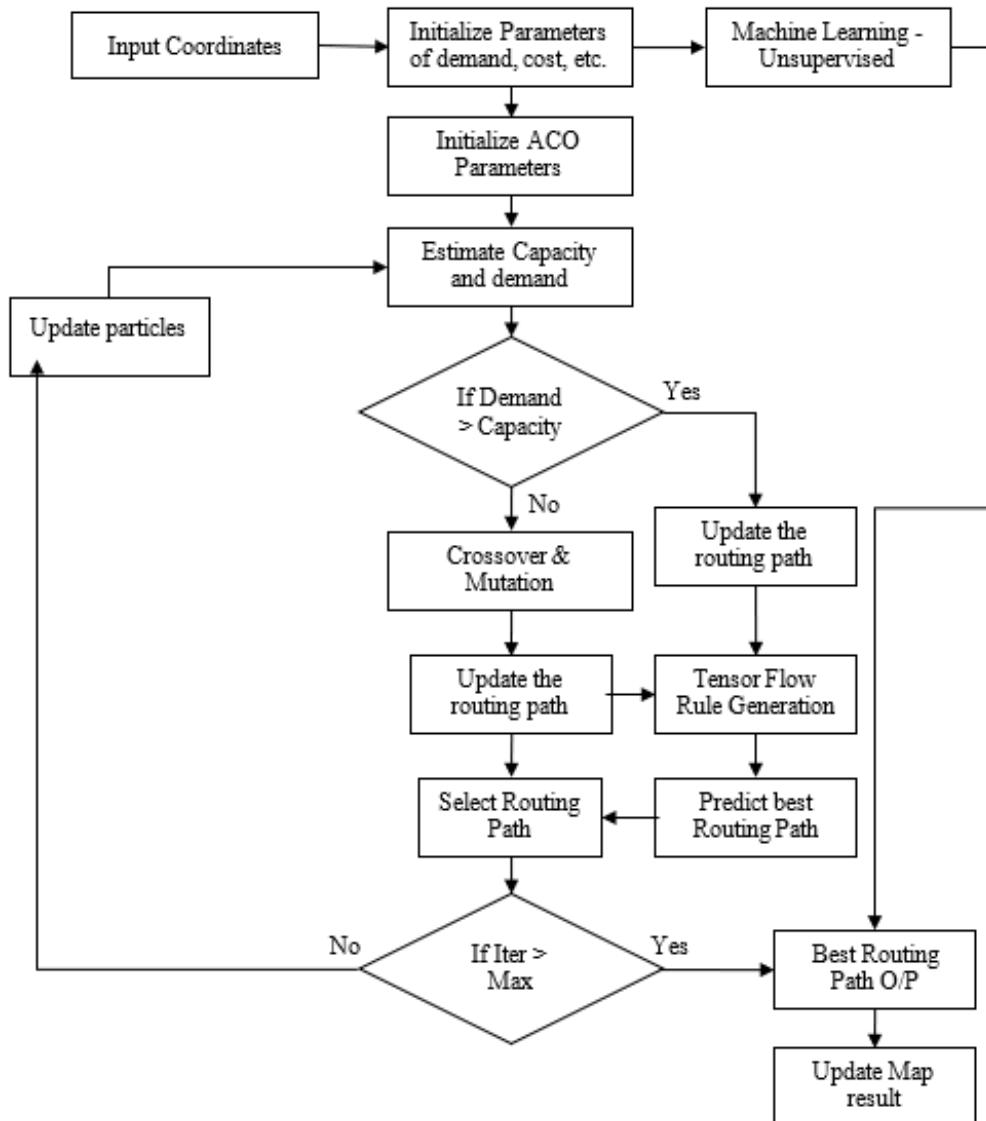


Fig 1. Proposed work

Once the parameters are fed, it combines with the ACO parameters, and two different genes are crossed. Here genes are nothing but destination routes. In this way the best parameter is mutated and fed into the channel till it yields the best route after multiple iterations. Lastly, The ACO-GA is integrated with google maps API. Once the ACO – GA results are provided, the results are calculated by TensorFlow model and the best route is decided, it verifies the current route with google maps and overwrites the best path on google maps which then updates the correct path.

3.1 Preprocessing

Data preprocessing or filtering is one of the most essential stage in data processing application systems, because the performance of classification system is highly dependent on the quality enhanced data. Filtering techniques are mainly used to remove noise/artifacts, improve contrast, improve quality, and smooth data. To this end, attribute collaboration and filtering techniques were used to remove noisy content present in the original data. This is one of the filtering approaches widely used in many data processing systems. The main reason for using this technique is to efficiently deblur the data with a better smoothing effect. Computes a filter function based on averaging neighboring pixels to remove noisy pixels. The primary reason for preprocessing the dataset is to filter the non-essential data. Here missing values which are the vehicle stops were inserted into he dataset values. Without merge values it could not be merged with google maps and performance cannot be measured without this. By merging the data with google maps, the data integration worked here. Data preprocessing also helped in selecting the best attributes from the Solomon benchmark dataset to improve performance.

3.2 Optimization

The optimization model of this paper is to focus on the hybrid model of ACO and GA optimization algorithm that are fused to perform the routing path selection for the vehicle database. In this the optimization helps in analyzing the cost value with the hybrid model of objective function to estimate the convergence of cost value for each iteration count. In this, the ACO selects the possible paths for the vehicle to travel in the way. The ACO-GA, when hybridized, provides the best path by performing the max number of iterations. When different paths are analyzed repeatedly, the algorithm chooses the shortest path with the least constraints and is chosen as the best path. Then, the GA estimates the acceptable level of the similarity and based on the cost value, the best path among the listed path was selected and justified as the best selection of routing path. Algorithm 1 explains the steps involved in hybrid ACO-GA optimization algorithm.

3.3 Tensor Flow Model

Following the extraction of the route patterns, the TF classification technique—which involves the processes of data training and testing—is used to precisely identify the

occluded item. The TF is developed based on the machine learning model of trained feature set, where the classification has been done by splitting the features into blocks. The main intention of this technique is processing the paths based on the separate blocks of featured paths, which helps to increase both the prediction accuracy and efficiency of classification. In this architecture, it has three layers mainly input, output and hidden layers where the input patterns are gained by using the input layer [21]. After this hidden layer starts processing the the patterns of path. Algorithm 1 explains the steps involved in tensor flow prediction algorithm. In the tenfor flow model, the matrix used is F1 score, precision , accuracy and recall.

3.4 Algorithm [3,7,23]

Input: Data $\{N_i\}$, flow link cost (FLC) c_{ij}

Output: Efficient routing selection process and security technique

For iteration = 1 to p

//Looping for 'p' number of data links in a database

Invoke γ be the data link's load value, which may be expressed

$$\gamma = \{N, L\}$$

// where, 'N' defines the number of database records and 'L' defines the distance between each record.

For i = 1 to n

//'n' is data count

For j = 1 to l

// 'l' is no. of connections.

Create network traffic $\{f_{i,j}^k\}$ for 'k' no. of attempts

$$\text{Update } \gamma_{i,j}^k = \gamma_{i,j}(f_{i,j}^k), \forall (i, j) \in L$$

Calculate route selection probability $\{y_{i,j}^k\}$ in the database design as

$$\{y_{i,j}^k\} = \sum_{s,d} \sum_t h_{s,d} \times P(r|C_n)(v_{s,d}(f_{i,j}^k)) \times a_{i,j}^r$$

Where, (s, d) represent the pair between source and destination.

$v_{s,d}$ – Vulnerability weight of docker path

$a_{i,j}^r$ – is the no. of transmissions to data in (i, j) as shown in the architecture for the territory of 'r'.

Update flow pattern,

$$f_{i,j}^{k+1} = f_{i,j}^k + \alpha_n \times (y_{i,j}^k - f_{i,j}^k)$$

Verify closure for every k+1 value

$$\text{Calculate } G(y) = \max(y_{i,j}^k).$$

// Find maximum potential point for determining selected duration

$$\text{Calculate } L(j) = \frac{1}{n} \sum_{x=1}^n \|N(f_{i,j}^k) - G(y)\|$$

//Find the distance vector between each data.

If $\lambda < L$, then

```

// λ defines the security strength of data
Data transmission.
    Calculate  $\Delta_{(j)} = \Delta_{j-1} + \mu \times \partial L / \partial W_i^l$ 
// From the data index, find the best data with the nearest distance property.
    Calculate  $c_{i,j} = W_j^l + \Delta_{(j)}$ 
// Adjust the vulnerability cost value
    Continue.
Else
    Affirm the path
    Use data parameters to evaluate risk parameters
        Continue loop.
    End If
End For 'j'
End For 'i'
From the updated table, pick the best option. ' $y_{i,j}^k$ ',
Update database weight and design architecture.
End For.

```

4 Results and Discussions

Here in this section, simulation results it discusses and the comparative analysis of ACO-GA with Tensor-Flow model and other optimization algorithm of the vehicle routing system. The overall test analysis was experimented with the standard dataset of Solomon's benchmark dataset which is publicly available and referred from the existing work of the routing system [26], there are 100 records taken for Solomon benchmark. The overall work was implemented in the PYTHON 3.8 tool in terms of '.py' scripts and related libraries. In this test analysis, the results can be compared with the other existing optimization methods that are referred from [22].

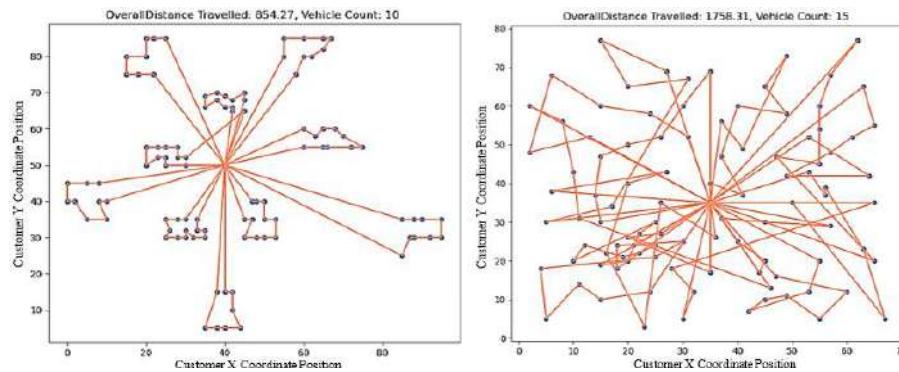


Fig. 2 and 3. Simulation Results for instance C101-100 Users and R105-100 Users

In that, the dataset contains the several position and coordinate information about the

user that are in the coverage area. In that, the dataset is divided into three different categories of ‘R’, ‘C’ and the combination of both as ‘RC’

Fig. 1 to 3 shows the simulation result for the sample of data combination of C201, R201 and RC201 for all the 100 number of users available in the dataset. In these graph result, the X and Y-axis represent the ‘X’ and ‘Y’ coordinate position of users respectively. Similarly, the fig. 4 shows the simulation result

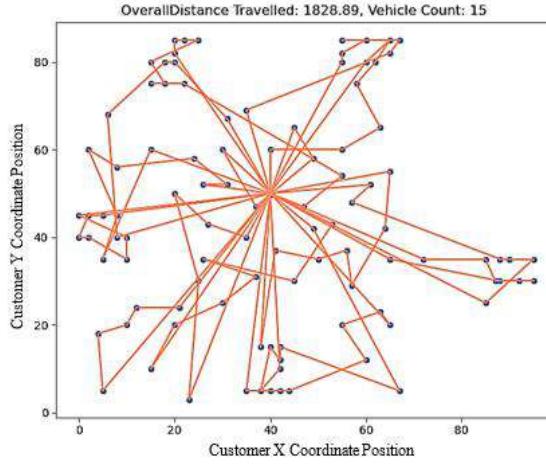


Fig. 4. Simulation Results for instance RC105-100 Users

This simulation step concentrates on the output graphical result of ACO-GA optimization for VRP. In that graph, the title displays the number of vehicles that are optimized and the amount of area that covered by those vehicles are mentioned for the 10th iteration of simulation result.

Table 1. Results of different parameters for proposed work.

Parameters	Value
ACC Macro	0.99326
Conditional Entropy	0.20469
Kappa	0.96153
Hamming Loss	0.02359
Overall ACC	0.97641
Overall MCC	0.96171
Overall RACC	0.38682
PPV (Macro-micro)	0.93204-0.97641
Reference Entropy	1.69515
TNR (Macro-micro)	0.99584-0.99607
Zero-one Loss	4449
Response Entropy	1.73724
Standard Error	0.00035
TPR (Macro-micro)	0.9769-0.97641

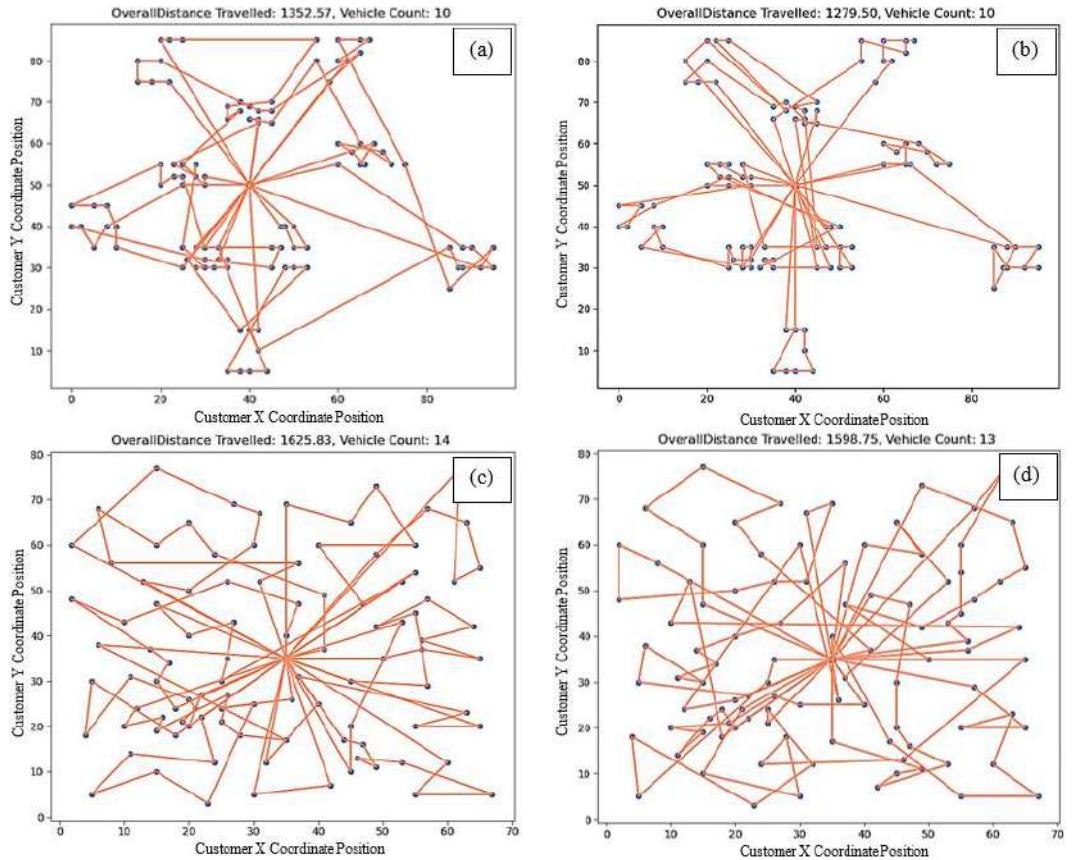


Fig. 5. Simulation plot for 100 users of instance (a) C103, (b) C104, (c) R103 and (d) R104

The maximum number of iteration was initialized as 100 to search for the 100 number of users that are to be linked. In that, the vehicle was started from the center of that are coverage and serves the link combination between users and the vehicle. Below in Table 1 it discusses about results of different parameters for proposed work and its values.

We can see the execution time taken for each simulation result is 5.491 seconds. We have multi-objective, along with cost, we have distance reduction etc. Cost function It is typically viewed as a single-purpose optimal solution that reduces the overall shipping costs of vehicles, which are the combination of their fixed costs plus expenses proportional to the distance travelled.

Figure 5 above shows for C103, C104, R103 and R104 cases. The above experiment uses Tensor flow(TF) model prediction for the results prediction of ACO-GA. The TF is created based on a machine learning model of a trained

functionality, in which the categorization is accomplished by slicing the features into blocks. The primary objective of this approach is to analyze pathways based on distinct blocks of featured paths, which improves both prediction accuracy and classification efficiency. The results show here have improved the Accuracy, Precision, Recall and F1 score [27] as shown in Table 2. The tensor flow This is because the Hybridization model performance is calculated from Tensor Flow model as the algorithm learnt from previous cases C101, R101 and measured the future responses.

Below in Table 2 it discusses Detection rate , this is used to predict the best routing path mentioned in figure 1 where it chooses the best routing path

Table 2. Detection Rate comparison of proposed vs. state of art methods.

Methods	F1 Score (%)	Accuracy (%)	Recall(%)	Precision(%)
PSO-SVM	19.3	48.8	14.8	27.9
ACO-SVM	23.1	72.9	15.7	43.3
GA-SVM	28.6	51.9	19.9	50.5
ACO-GA	73.2	67.3	75.2	71.3
Neural Network	76.8	74.7	76.3	77.3
Proposed	83.6	81.4	84.2	84.3
ANN	74.2	71.9	75.4	73
PNN	76.8	74.7	76.3	77.3
Proposed	83.6	81.4	84.2	84.3

In the above table we have used C,R,RC which represent cases . Here the data sets have been used from Solomon benchmark and is usedfor 100 customers for all cases.

5 Conclusion

The paper work proposed the Tensor Flow based vehicle routing prediction along with the optimal routing path selection of ACO-GA. By referring to the parameters of path distance and the number of vehicles that need to traverse the region, the routing path is generated from the data of vehicle coordinates and the delivery point. This also reduces the fuel cost and the time consumption of overall system. The Tensor Flow predicts the path that are best along with ACO-GA by estimating the cluster of the relevancy that are trained with the feature patterns of data. The experimental result and the graph shows the traveling distance and the amount of area that the vehicle can travel with minimum number of vehicle count. These type of TF based VRP system improves the prediction performance and the process of optimal path selection compare to other existing model of VRP.

For future papers we can implement optimized VRP paper for different structure of data combination and can integrate with the AI model to update the training model based on the feature update from time instant.

References

1. Qiu, Meng, et al. "A Tabu Search algorithm for the vehicle routing problem with discrete split deliveries and pickups." *Computers & Operations Research* 100 (2018): 102-116.
2. Min, Jianing, Cheng Jin, and Lijun Lu. "A three-stage approach for split delivery vehicle routing problem solving." 2018 8th International Conference on Logistics, Informatics and Service Sciences (LISS). IEEE, 2018.
3. Koç, Çağrı, and Gilbert Laporte. "Vehicle routing with backhauls: Review and research perspectives." *Computers & Operations Research* 91 (2018): 79-91.
4. Neves-Moreira, Fábio, et al. "The time window assignment vehicle routing problem with product dependent deliveries." *Transportation Research Part E: Logistics and Transportation Review* 116 (2018): 163-183.
5. Madankumar, Sakthivel, and Chandrasekharan Rajendran. "Mathematical models for green vehicle routing problems with pickup and delivery: A case of semiconductor supply chain." *Computers & Operations Research* 89 (2018): 183-192.
6. Gschwind, Timo, Nicola Bianchessi, and Stefan Irnich. "Stabilized branch-price-and-cut for the commodity-constrained split delivery vehicle routing problem." *European Journal of Operational Research* 278.1 (2019): 91-104.
7. Gayialis, Sotiris P., Grigorios D. Konstantakopoulos, and Ilias P. Tatsiopoulos. "Vehicle routing problem for urban freight transportation: A review of the recent literature." *Operational research in the digital era-ICT challenges* (2019): 89-104.
8. Sitek, Paweł, and Jarosław Wikarek. "Capacitated vehicle routing problem with pick-up and alternative delivery (CVRPPAD): model and implementation using hybrid approach." *Annals of Operations Research* 273.1 (2019): 257-277.
9. Liu, Dan, et al. "Two-echelon vehicle-routing problem: optimization of autonomous delivery vehicle-assisted E-grocery distribution." *IEEE Access* 8 (2020): 108705-108719.
10. Hornstra, Richard P., et al. "The vehicle routing problem with simultaneous pickup and delivery and handling costs." *Computers & Operations Research* 115 (2020): 104858
11. Li, Jiliu, et al. "Branch-and-price-and-cut for the synchronized vehicle routing problem with split delivery, proportional service time and multiple time windows." *Transportation Research Part E: Logistics and Transportation Review* 140 (2020): 101955.
12. Wang, Yong, et al. "Collaboration and resource sharing in the multidepot multiperiod vehicle routing problem with pickups and deliveries." *Sustainability* 12.15 (2020): 5966.
13. Marques, Alexandra, et al. "Integrated planning of inbound and outbound logistics with a Rich Vehicle Routing Problem with backhauls." *Omega* 92 (2020): 102172.
14. Dumez, Dorian, Fabien Lehuédé, and Olivier Péton. "A large neighborhood search approach to the vehicle routing problem with delivery options." *Transportation Research Part B: Methodological* 144 (2021): 103-132.
15. Park, Hyungbin, et al. "Waiting strategy for the vehicle routing problem with simultaneous pickup and delivery using genetic algorithm." *Expert Systems with Applications* 165 (2021): 113959.
16. Euchi, Jalel, and Abdeljawed Sadok. "Hybrid genetic-sweep algorithm to solve the vehicle routing problem with drones." *Physical Communication* 44 (2021): 101236.
17. Liu, Wenjie, et al. "A hybrid ACS-VM algorithm for the vehicle routing problem with simultaneous delivery & pickup and real-time traffic condition." *Computers & Industrial Engineering* 162 (2021): 107747.
18. Balasubramaniam, Vivekanandam. Design an Adaptive Hybrid Approach for Genetic Algorithm to Detect Effective Malware Detection in Android Division. *Journal of Ubiquitous Computing and Communication Technologies*. 3. 135-149 (2021).
19. Konstantakopoulos, Grigorios D., Sotiris P. Gayialis, and Evripidis P. Kechagias. "Vehicle routing problem and related algorithms for logistics distribution: A literature review and

- classification." Operational research 22.3 (2022): 2033-2062.
20. Huang, Shan-Huen, et al. "Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm." Advanced Engineering Informatics 51 (2022): 101536.
 21. International Conference on Automation and Computing Yu H. Chinese Automation and Computing Society in the UK University of Lancaster & Institute of Electrical and Electronics Engineers. Improving productivity through automation and computing : 2019 25th international conference on automation & computing : lancaster university uk IEEE (2019).
 22. Samra, Bouakkaz & FouziSemchedine. A certificateless ring signature scheme with batch verification for applications in VANET. Journal of Information Security and Applications. 55 (2020). 102669.
 23. Min, Jianing, Lijun Lu, and Cheng Jin. "A Two-Stage Heuristic Approach for Split Vehicle Routing Problem with Deliveries and Pickups." LISS 2021. Springer, Singapore, (2022). 478-490.
 24. Chowlur Revanna, Jai Keerthy & Al Nakash,Nushwan.Vehicle Routing Problem with Time Window Constrain using KMeans Clustering to Obtain the Closest Customer. Global Journal of Computer Science and Technology. GJCST, vol. 22, no. D1, pp. 25–37. 10.34257/GJCSTDVOL22IS1PG49.(2002)
 25. Chowlur Revanna, Jai Keerthy & Rakesh Veerabhadrappa . “Analysis of Optimal Design Model in Vehicle Routing Problem based on Hybrid Optimization Algorithm” . 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N-22).(2022).
 26. Solomon, Marius M. "Algorithms for the vehicle routing and scheduling problems with time window constraints." Operations research 35.2 (1987): 254-265.
 27. Sokolova, M., Japkowicz, N., & Szkakowicz, S. December). Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In Australasian joint conference on artificial intelligence (pp. 1015-1021). Springer, Berlin, Heidelberg (2006) .

Authentication key generator for data sharing on cloud- A Review

Santhosh Krishna B V, B Rajalakshmi ,Esikala Nithish Manikrishna, Gundre
SaiSruthi,Gangireddy Ramyasri, Ashok K

New Horizon College of Engineering Bengaluru, Karnataka
santhoshkrishna1987@gmail.com

New Horizon College of Engineering Bengaluru, Karnataka
hod_cse@newhorizon india.edu

New Horizon College of Engineering Bengaluru,Karnataka
1nh19cs047.nithish@gmail.com

New Horizon College of Engineering Bengaluru,Karnataka
1nh19cs056.saisruthigundre@gmail.com

New Horizon College of Engineering Bengaluru,Karanataka
ramya.1nh19cs053.bsec@gmail.com

New Horizon College of Engineering Bengaluru, Karnataka
kashok16@gmail.com

Abstract. With the help of a sizable quantity of virtual storage, cloud computing provides services through the Internet on demand. The primary benefit of cloud computing is that it relieves users of the need to invest in pricey computer equipment. Lower expenses are related with infrastructure. Researchers are looking at new, relevant technologies as a result of recent breakthroughs in cloud computing and other sectors. Due to its accessibility and scalability for computer operations, both private users and companies upload their software, data and services into the cloud storage. While switching from local to remote computing provides advantages, there are also a number of security issues and difficulties for both the supplier and the customer. There are many cloud services offered by reputable third parties, which is raising security concerns. The cloud service provider offers its services over the Internet and makes use of numerous online technologies, which raises fresh security concerns. One of the most important requirements for today is online data exchange for greater productivity and efficiency. In this work, we have done a detailed analysis on cloud based security issues and possible solutions. with this data owners can save and share the data through online.

Keywords: cloud protection, Framework for cloud computing, security issues, threats and attacks

1 Introduction

With the recent emergence of cloud computing, countless applications that span international boundaries and include millions of users have seen their ability to share data stretched to the limit. Today, governments and businesses view data

sharing as a crucial instrument for increased efficiency. Social networking, healthcare, and education have all been transformed through cloud computing. The capacity of cloud computing to enable global data sharing and exchange between numerous users without the pains of manual data transfers and without the production of redundant or outdated documents may be its most intriguing use case. Social networking sites have made the globe more interconnected by utilizing the cloud to enable the sharing of text and multimedia. Cloud platforms frequently feature collaborative tools, which are very well-liked since they increase productivity and effort synchronization. The result of cloud computing has spread all over to the healthcare industry as well, with mobile applications enabling remote patient monitoring. In summary, cloud computing is drastically altering many facets of our existence. Despite all of its benefits, the cloud is vulnerable to security and privacy breaches, which pose a serious obstacle to its widespread adoption as the main method of data sharing in today's society. In a poll conducted by [1] cloud customers ranked security as their top difficulty, with 75% of people concerned about the security of their vital IT and business systems. Unknown service providers also be considered, notwithstanding the prevalence of security risks from outside agents. It is not simple to guarantee security and privacy on the cloud because online data almost always occupy in shared environments (for example, many virtual computers running on the same physical device). Setting down the specifications that a data sharing service must meet in order to be considered secure is crucial when discussing the protection of data privacy and protection in the cloud.

2 THE ARCHITECTURAL FRAMEWORK FOR CLOUD COMPUTING:

The fundamental architectural framework for cloud computing is provided. The fundamental idea and architecture of cloud computing must first be understood in order to comprehend the security concerns. The widely used NIST provides four deployment models, five basic characteristics, and three service delivery models [2].

2.1 Fundamental qualities

There are many attributes of cloud computing and these key attributes makes the technology in demand, the key attributes are[2]:

2.2 On Demand self service

It allows customers to use web services and administration interfaces to direct request, manages, and access services without interacting with any human beings.

2.3 Access to a large network

Any standard device, including smartphones, PCs, desktop computers, and laptops, must be able to access data and services provided in cloud. These devices operate using some common technology and protocols. Because of its nature, cloud computing ought to accommodate all established protocols.

2.4 Resource pooling

Large physical or virtual computer resources are made available by the cloud provider and are distributed among numerous consumers. In a multi-tenant setting, these resources are assigned in a dynamic manner.

2.5 Rapid Elasticity

One crucial characteristic of the cloud is elasticity. The resources used for this property are scaled based on consumer needs. Customers have infinite resources that they can pay for on a pay-per-use basis as needed.

2.6 Measured Service

The cloud system's metering functionality allows the resources to be automatically controlled and scaled in accordance with user demand and paid services.

2.7 Service paradigms

A set of services is supplied by the service model; the consumer uses these services, which are given by the provider.

2.8 SaaS

The SaaS gives its clients online software services to access the applications and transfer the details related to the work and programmes to remote storage servers (IDE). Salesforce and customer relationship management are two examples that match the SaaS paradigm.

2.9 IAAS

The term "IaaS" refers to virtualizes resource, such as compute, storage, network, memory, and processor, that the cloud service provider makes ad hoc and on-demand available. The finest IaaS example is Amazon Web Service [3], which offered EC2 services like that are virtual machines with a software stack.

2.10 PAAS

PAAS, a more complicated programmable platform, is offered by the platform-oriented cloud. Cloud users can utilize a variety of models, an IDE, specialized services, os and platform-level resources on an easily programmable cloud platform to design, run, deploy, and manage their applications.

2.11 Cloud deployment of private

A company or a TPA service internally controls and manages an exclusive cloud.

2.12 Cloud deployment of public

The CSP operates and manages a public cloud, and the user's off-site location may host the actual infrastructure. The resources in the cloud are shared by many users, who pay the cloud provider for the services they utilize.

2.13 Hybrid Cloud

Two or more clouds with the same architecture and capabilities can be combined to create a hybrid cloud.

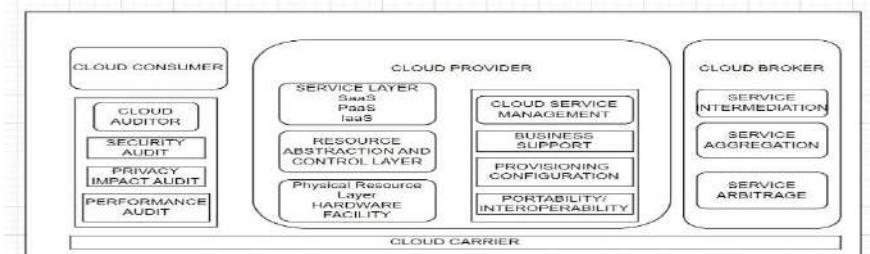


Fig. 1. The architectural framework for cloud computing

3 METHODS OF CLOUD SECURITY ISSUES

3.1 Storing of data in cloud security

Since the cloud computing paradigm does not permit the control over the data and makes it difficult to verify data integrity and confidentiality, loss of control in the storage problem is a severe concern. The user of cloud computing is geographically separated from the server that stores their data and does processing. A network of computers for storing cloud data is made available by using cloud computing. The cloud service provider is in charge of and oversees the server pool, whose location

is Unknown. The abstraction of the virtual layer makes finding the real layer more challenging. Data pooling, data locality several locations, remote data storage, loss of control, and sophisticated models for integrity verifying the solution are some examples of security challenges with data storage.

3.2 Untrusted computing

The front end interface for SaaS applications is often what security services want to provide when users request a web service or an HTML page. These programmes can be modified or adjusted to fit a certain pattern of behaviour. The session state manager, extra services, and any possible reference data that the request may call make up this pattern of behaviour. The request is simply forwarded from one service to another, and so on, building a service tree whenever one application or service calls another. Due to server misconfiguration and criminal behaviour, a computing framework which computes enormous data sets in distributed systems may provide the unintended, unreliable, and dishonest conclusion. Security issues with untrusted computing are Top-down SLAs. Dishonest computing, malicious users, outages, slowness, and others[5] Inadequate computer model security measures, root-level backup errors[10], migration and restoration challenges, sluggishness and outages Data and service accessibility utilising phoney resources[11].

3.3 Service and data availability

The real and virtual resources of the cloud's database and processing servers are extremely accessible. At the application and infrastructure levels, architectural modifications are required to provide high availability and scalability of services and data. Running applications on many servers is one strategy. This technique facilitates DoS assaults. This method has the advantage of providing a backup application server in the event that the primary one fails, ensuring the availability of data and services. The server could also be working on a particularly demanding application job, in which case he will consume more power, resources, and time. The cost of further calculations and application availability are probably increasing as a result Counterfeit resource utilisation and Cloud disruption are security concerns with data and service availability [12,13,14].

3.4 Cryptography

Information and data stored in the cloud are protected using cryptographic techniques. The concept behind achieving cloud security is simple. It changes standard text into cypher text and other types of text. The idea is based on the assumption that it would be difficult to determine the value of the plain text data in the case that a cypher text was accessible. They must carefully and robustly construct cryptography systems since the encryption key is what determines the

overall level of security. The prime factorization of big integers makes the Rivest Shamir Adleman (RSA)-based encryption more safe. Hardware availability is one of cryptography's security concerns (hardware fault) Ineffective key management [15], flawed cryptography algorithms, brute force, and dictionary attacks are all examples of insecure cryptography mechanisms.[17]

3.5 Data recycling

Reusing the cloud space after the data had been effectively used and disposed of was a smart idea. But the next user must be prevented from accessing the data that was used by the preceding user. Sanitization is the process of clearing out or eliminating specific pieces of data from a resource. People can access updated data in a dispersed manner following sanitization. Data sanitization is a critical activity in distributed systems because it allows for the proper data disposal and data selection when data is sent to the trash. Because the hard drive can be erasing some crucial data, incorrect sanitization leads to data leakage and loss. The ineffective application of data destruction policies [18], the disposal of unused hard drives [14], the use of hard discs by many tenants [15], and resource recycling [19,20,15] are security concerns with cloud data recycling.

3.6 Malware

It is active and performs activities at every three minutes at a single business. An online data storage system is MediaFire and SugarSync cloud-based service provider generates a distinct security issue they either copy the features and data properties of the many devices. The primary issue there is that if one system contains then, because of inheritance, the malware spreads throughout the cloud. Malware also poses a serious threat to cloud devices so that it can be used to corrupt or erase the details.

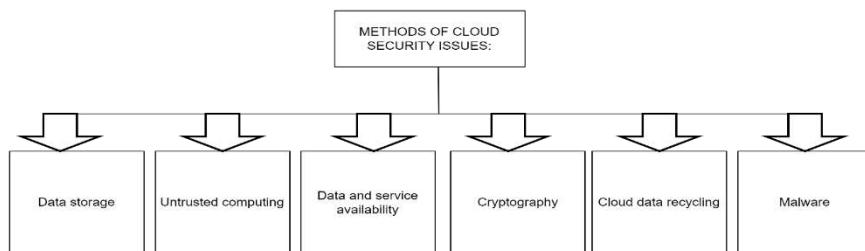


Fig. 2. Methods of cloud security issues

4 CLOUD PROTECTION

Computer security includes cloud security. It specifies a system of regulations, controls, and technologies that are useful for safeguarding data and services. Threats and attacks have an effect on the cloud system. New security issues might arise as a result of the cloud resources' integrity, availability, and confidentiality being compromised, as well as the services offered at different levels. In this part, we'll look at a few security concepts in order to have a better knowledge of cloud issues.

4.1 Concepts for security of cloud

Numerous security problems and risks are addressed by cloud security. To assist readers comprehend the idea of cloud security, the study emphasises the sources of the risk and threats. To better understand the security challenges which are prevalent in the cloud, this section examines a few cloud-specific topics, including virtualization, multitenancy, cloud platforms, data outsourcing and trust management.

4.2 A virtualization-related issue

The idea of virtualization is the division of services, programmes, computing resources, and operating systems from the actual physical hardware on which they are executed. The virtual machine (VM) and virtual machine manager are elements of virtualization (VMM).

4.3 Multi-tenancy

It is a feature of the computing environment that enables one or more tenants to share each operating process, introduces the concept of sharing instance. It gives consumers the choice to share a single cloud platform with other users. Consider an IaaS company. A VMM is a platform for multi-tenancy sharing, and instances are referred to as VMs.

4.4 Threat agents

Threat agents are a type of entity that can fend off an attack and create threats. Whether it occurs internally or externally, this threat is caused by a person or any software programme. A network level attack is sent by an external software programme or external person known as an anonymous attacker via a public network. Those who live in the cloud environment without the administrator's authorization are considered untrusted cloud service users.

5 THREATS TO CLOUD COMPUTING

Anything that has the potential to seriously harm a computer system is considered a threat in the context of computer security. Threats might lead to potential attacks on the system or network infrastructure.

5.1 Different service delivery

Both the business model and the cloud computing model use unique methods for delivering and receiving services. Therefore, cloud computing has the flexibility to alter its own method of service delivery. The firm must evaluate all the risks associated with losing control of the cloud because the cloud service provider has transferred all services and applications to a remote location. When sending cloud data between two locations, security rules specific to each location must be followed. This is one of the key risks that are brought about by using something. To eliminate such threats, one needs robust end-to-end encryption, globally recognized security regulations, and a trust application.

5.2 Misuse and criminal application of cloud computing

Provision of information These utilities are provided by vendors, including boundless network, storage, and bandwidth. Some service providers let customers try out their products for a set amount of time. This is typically used in conjunction with a simple registration process that enables anybody to register and access cloud services without having to go through a security process. They now lack the power to make an impact on the user for the duration of the trial. Additional potential concerns include the hosting of potentially harmful materials, password and key finding, captcha solving , and distributed denial of service (DDoS) assaults. Spammers, creators of harmful programs, and other criminals may now carry out their assault. These weaknesses endanger the infrastructure for PAAS and IAAS.

5.3 Unsecured software interfaces and APIs

Customers can utilise a number of software interfaces and APIs that the cloud provider offers to communicate with cloud services. The complexity of the cloud is increased by the interfaces' layer-like deployment on top of the cloud base. These interfaces offer their customers full provisioning, management, and monitoring capabilities. As a result, the security of these APIs is crucial to the cloud's availability and security. But occasionally, both deliberate and unintentional attempts can compromised the security to these APIs. PaaS, IaaS, and SaaS service models may be impacted by these kinds of API attacks. As other parties frequently use these interfaces to deliver services, there is also the possibility of another form of risk.

5.4 Malicious insiders

Malicious software one of the main dangers of cloud computing. internal dangers, as a result of many organisations' lack of details regarding her access level and employee hiring process for their workers of internal resources. Mostly, this threat is carried out because of the clients' use of IT services, lack of transparency, and collaborating inside a single management domain. Somehow, a worker a greater amount of access as a result, the confidentiality of Services and data are compromised. This also leads to a circumstance in which an insider attacker could acquire sensitive information and impact the cloud.

5.5 Issues with shared technology in a multi-tenancy setting

Information as a service providers leverage the virtualization idea to deliver the services in a multi-tenant environment. Through virtualization, it is feasible for more users to share the resource. Hypervisor in the multi-tenant system could divulge user information to an unauthorised user. Because the infrastructure was not designed to effectively offer isolation in a multitenant context, there is a significant risk involved. Sharing could affect the cloud architecture as a whole by enabling single user to access data over another user. Access restriction and stronger authentication are two strategies for resolving this problem.

5.6 Data leakage and loss

The collaborative and productive nature of computing in cloud, data loss can also result from the loss of an encoding key. Theft, modification, and Examples of data loss include data deletion without a backup of the original data. The primary reasons of data loss and leakage include weak encryption methods, weak keys, association risk, unstable data centres, and a lack of disaster recovery. Weak access control, authorization, and authentication are other contributing causes. All service model types are vulnerable to these dangers. Safe APIs, data backups, powerful encryption keys, secure storage, and data integrity are a few prevention strategies.

5.7 Hijack service

The customer may be forcibly steered to a risky website during the service hijacking process. Fraud, phishing, and the usage of software bugs are all methods that can be used to accomplish this. Reusing login credentials and passwords frequently results in these attacks. In cloud computing, if a hacker gets hold of someone's login information, they can record actions, alter data, provide false information back to the customer, or direct the client to hacked accounts and unauthorised websites.

5.8 Risk profiling

Cloud services are less involved in owning and maintaining infrastructure and software because of the high workload. The cloud offers contracts to businesses for the upkeep of their software and infrastructure. This ignorance leads to greater threats and dangers. The cloud should have a mechanism in place for keeping an eye on and making changes to logs, data, and infrastructure-related information in order to eliminate threats.

5.9 Theft of Identity

Theft is variety of fraud in which a perpetrator uses another person's name, credentials, resources, or other service benefits in order to get access to protected information. The victim suffers several unpleasant consequences and financial loss as a result of these threats. Keyloggers, phishing scams, and inefficient password recovery techniques, among other things, may all contribute to this danger. The security idea includes both strong multi-tier authentication techniques and a trustworthy password recovery procedure.

THREATS:

Table 1. Different types of threats, their effects and solutions

Threats	Effects	Solutions
Different service delivery	Loss of control over the infrastructure of the cloud	provided services that were controlled and supervised
Misuse and criminal application of cloud computing	Due to unclear sign-ups, there is a loss of validation, service fraud, and a stronger attack.	Observe the state of the network and use strong registration, authentication methods.
Unsecured software interfaces and APIs:	Incorrect transfer of the content, improper, authentication, and authorization	Strong access control and authentication measures are used, and data transfer is secured.
Malicious insiders	resource penetration, asset damage, productivity loss, and operational impact	Utilization reporting and breach alerts, as well as open security and management procedures
Issues with shared technology in a multi-tenancy setting:	By exploiting the hypervisor, interfere with one user service and other user services.	Audit configuration and vulnerabilities, and utilise strong authentication and access control procedures for administrative tasks.
Data leakage and loss	Data that is personally sensitive may be altered, destroyed, damaged, or erased.	provide systems for data backup and storage
Service hijacking	Stolen user account credentials give access to a crucial region of the cloud, putting the security of the services at risk.	use of powerful authentication techniques, security guidelines, and encrypted communication
Risk profiling	Operations involving internal security, security guidelines, configuration breaches, patching, auditing, and logging	Recognize incomplete logs, infrastructure, and data aspects in order to safeguard the data use monitoring and altering system.
Identity theft	To access that user's resources and obtain credits or other benefits under that user name, an aggressor can obtain the identity of a legitimate user.	Authentication methods and strong multi-tier passwords should be used.

ATTACKS:

Table 2. Different types of attacks, their effects and solutions

Attacks	Effects	Solutions
Zombie attack	Affected service availability; possibility of creating a phony service	robust authorization and authentication
attack using service injection	Service integrity is compromised, and users are given malicious services in place of legitimate services.	Service integrity is compromised, and users are given malicious services in place of legitimate services.
port checking	Unusual service behavior reduces service availability	Strong port security is necessary
Phishing attack	Affect the user's private information that shouldn't be shared	employee a secure web link (HTTPS)
attack through the backdoor	has an impact on the service's accessibility and data privacy, and offers rights for accessing legitimate user resources.	Strong authentication, identification, and isolation procedures are necessary.

CONCLUSION

Cloud computing include rapid system implementation, low costs, abundant storage, and simple system access from anywhere at any time. As a result, cloud computing is becoming more and more apparent in recent technical developments and a widely utilised computer environment everywhere. Several security and privacy issues make it difficult to use cloud computing. The security weaknesses, dangers, and assaults that the cloud already has should be known to all users. If businesses are aware of security threats and attacks, they can adopt the cloud more swiftly.. Utilizing both traditional and cutting-edge techniques and technology, cloud computing. Multiple clouds can be produced using this innovative technique. Particular security concerns utilise the same resources physically located at the cloud from several places to virtualization and multi-tenancy features. The security of the system may be hampered by improperly segregated VMs. We can increase cloud security and lessen risks and assaults by creating the keys.

References

- [1] IDC Enterprise Panel. "It cloud services user survey, What users want from cloud services providers", august 2008.
- [2] M. O'Neill, "NIST: The NIST Definition of Cloud Computing" Accessed 2013 September.
- [3] "Amazon: Amazon Web Services: Overview of Security Processes", Accessed 2015 November.
- [4] P. Syam Kumar and R. Subramanian, " An efficient and secure protocol for ensuring data storage security in Cloud Computing",2011, IJCSI Int. J. Comput. Sci. Issues 8 6, 8.
- [5] B. V. Santhosh Krishna, S. Sharma, K. Devika, Y. Sahana, K. N. Sharanya and C. Indraja, "Review of Fake Product Review Detection Techniques," 2022 Second International

Conference on Artificial Intelligence and Smart Energy (ICAIS), 2022, pp. 771-776, doi: 10.1109/ICAIS53314.2022.9742735.

[6] S. Bangari, P. Rachana, N. Gupta, P. S. Sudi and K. K. Baniya, "A Survey on Disease Detection of a potato Leaf Using CNN," 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), 2022, pp. 144-149, doi: 10.1109/ICAIS53314.2022.9742963.

[7] M. Zhou, R. Zhang, W. Xie, W. Qian and A. Zhou, " Security and privacy in cloud computing: A survey. In: Semantics Knowledge and Grid ", 2010 Proceedings of the Sixth International Conference on Nov 1 pp. 105–112. IEEE.

[8] S. Subashini and V. Kavitha, " A survey on security issues in service delivery models of cloud computing" J. Netw. Comput. Appl. 34 1, 1–11,2011.

[9] B.Meenakshi Sundaram; B Rajalakshmi; Babu Aman Singh; Rachit S Kumar; Rohith Arsha, Disaster Relief Compensation Computational Framework, 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), DOI: 10.1109/ICAIS53314.2022.9742829

[10] Uma.N,Prashanth CSR,A detailed analysis of the various frequent itemset mining algorithmsJournal of Advanced Research in Dynamical and Control Systems, 2020, 12(2 Special Issue), pp. 448–454

[11] C.Rong, S.T. Nguyen and M.G.Jaatun, "Beyond lightning: a survey on security challenges in cloud computing", Comput. Electr. Eng. 39 1, 47–54,2013.

[12] B.P.Rimal, A. Jukan, D. Katsaros and Y. Goeleven, " Architectural requirements for cloud computing systems: an enterprise cloud approach", J. Grid Comput. 9 1, 3–26,2011.

[13] Alpha Vijayan; B. Meenaskhi; Aditya Pandey; Akshat Patel; Arohi Jain, Video Anomaly Detection in Surveillance Cameras, 2022 International Conference for Advancement in Technology (ICONAT), DOI: 10.1109/ICONAT53423.2022.9726078

[14] M.Armbrust, A.Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee,D. Patterson,A. Rabkin, I.Stoica and M. Zaharia, " A view of cloud computing", Commun. ACM 53 4, 50–58, Apr 1,2010.

[15] K. Ashok, A. B. Gurulakshmi, M. B. Prakash, R. Poornima, N. S. Sneha and V. Gowtham, "A Survey on Design and Application Approaches in Women-Safety Systems," 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022, pp. 101-110, doi: 10.1109/ICACCS54159.2022.9784981.

[16] Muqeem Ahmed, Mohd Dilshad Ansari, Ninni Singh, Vinit Kumar Gunjan, Santhosh Krishna B. V., Mudassir Khan, "Rating-Based Recommender System Based on Textual Reviews Using IoT Smart Devices", Mobile Information Systems, vol. 2022, Article ID 2854741, 18 pages, 2022. <https://doi.org/10.1155/2022/2854741>

[17] P.Thorsheim, "The Final Word on the LinkedIn Leak", accessed September 2015.

[18] P.A. Boampong and L.A. Wahsheh, "Different facets of security in the cloud", In: Proceedings of the 15th Communications and Networking Simulation Symposium, Mar 26, Society for Computer Simulation International. p. 5,2012.

[19] A. Casale, " The Dangers of Recycling in the Cloud",2013.

[20] S.Pearson, " Privacy, security and trust in cloud computing", Jan 1. Springer London,pp. 3–42,2013

Restaurant quality analysis: A machine learning approach

* Rohit B. Diwane 1, Dr. Kavita S. Oza 2, Dr. Varsha P. Desai 3

1, 2 Department of Computer Science, Shivaji University Kolhapur, Maharashtra, India.

^{1*} rohitdiwane111@gmail.com, ² kso_csd@unishivaji.ac.in

3 Department of Computer Studies, V.P. Institute of Management Studies & Research, Sangli.

³ varshadesai9@gmail.com

Abstract. Managing customer's happiness has emerged as a significant business trend, particularly in the restaurant industry. The purpose of this study is to determine how K-Means algorithms can be used to measure customer satisfaction at a family restaurant in Kolhapur. A survey is carried out related to services and ambiguous at the restaurant. What makes restaurants popular is the main focus of the survey. Data collected through online survey is clustered using the elbow method as well as the K-Means clustering. This study presents the results of the customer satisfaction measurement and offers improvement and recommendations to the concerned restaurant.

Keywords: K-Means Algorithms, Elbow method, Principal Component Analysis, Clustering, Restaurant.

1. Introduction

The restaurant sector is among the most important economic contributors in the Asia. Asian-style restaurants can be found all over the world, from small, cheap street vendors to luxurious, expensive hotels and shopping areas [1]. Almost every day, a large variety of new food items are released. In this regard, the restaurants depend on the qualities of customer service needed to be consistently maintained in order to stand out in the severe business competitions. preferences for services among very specific demographic groups [2].

People from various cultures like going to different restaurants. The restaurant recommendation system is currently the most important recommendation system in the modern world, where all families visit various restaurants to spend some time together peacefully. For a family, a single individual, or a couple to pick where to go based on convenience, the restaurant recommendation is essential [3].

For many various sorts of businesses, especially restaurants, managing customer happiness has emerged as a major issue. This would allow the restaurant to identify the area that needs improvement and would keep them competitive itself. The quality of the meal, the price, and many other factors can all be used to measure consumer happiness [4]. Because it will have an impact on the analysis' results, the choice of features is also crucial in managing customer happiness. When the purpose is to organise data samples into clusters based on some measure of similarity, this procedure is known as clustering. Clustering differs from classification in that it is performed without supervision, therefore no class labels are given, and sometimes even the number of clusters is unknown beforehand [13].

The primary goal of clustering is to group comparable data points together. Many applications in data analysis and data visualisation depend on having well-defined clusters of data points. Researchers today have the option of using data clustering to analyse and extract information from the data. With the aid of the numerous algorithms employed in data clustering, researchers can identify patterns in the data, analyse it, and predict a phenomenon. This approach has been used to address a wide range of daily issues in sectors like restaurants, pharmaceutical, management, and safety. Because raw input databases are used by data clustering systems, this poses issues. For instance, databases frequently exhibit volatile, partial, noisy, and large characteristics. Implementing data clustering difficulties requires some incomplete information, misfits, and prediction accuracy [5] [12].

In this survey, we investigated the customer service in restaurants. using the clustering approach in a Kolhapur restaurant to cluster data. We have used clustering to read patterns of customer feedback depending on some measurement data about the restaurant. PCA, or Principal Component Analysis was used in high-dimensional data (no. of questions) should be placed into smaller dimension. Additionally, we looked for a parameter that played a significant impact on the outcomes. This work is really important. because we're committed to preserving customer service rarely do in our daily lives. We evaluated the level of customer instruction given and made recommendations for improvement to the restaurant.

2. Review of Literature

Research shows that essentially, the work in this paper focuses on the usage of SVM as a machine learning algorithm along with implementation of this method utilising decision trees, K-nearest neighbours, naïve bayes, and random forests. This research uses a machine learning strategy to analyse and categorise

restaurants using sentiment. The methodology used in this work included data pre-processing, word preparation, training and test data segmentation, implementation, and performance analysis. The study's analysis of restaurants with the best performance is done utilising decision trees and SVM techniques [6].

Agglomerative clustering, customer satisfaction, data mining, K-means, and spectral clustering algorithms are all discussed in this research as they relate to customer happiness in restaurants. In a paper, prediction using clustering and data mining are used as the framework for the study. Data collection, pre-processing, and a clustering model are all part of the procedure. The major interest is on collecting information from the restaurant and creating clusters to meet consumer expectations in the restaurant [7].

This research uses machine learning, random forest, and decision tree techniques to examine the literature on tree-based machine learning. According to the paper, restaurants are divided into various classes based on their service standards. In the study, random forest and decision tree both go through the same data collection, pre-processing, ML algorithm application, interpretation, and data visualisation processes. The primary concern is comparing decision tree and random forest efficiency in terms of class prediction [8].

This paper had work on sentimental analysis for restaurant recommender. However, analysis perform static information like price, qualities and services. Also, in semantic approach used for making cluster. Methodology used as pre-processing, clustering, sentiment analysis is followed. The research of main focus is finding comments of restaurant from online. Using natural language processing extract the qualities of that restaurants. The clustering results on Wu-palmer method carried out highest accuracy [9].

A better strategy to maximise efficiency and ensure that everyone has access to nutritious food is to adopt sustainability. In this study, the primary focus is on restaurant reviews using Fuzzy Domain Ontology (FDO) and Support Vector Machines (SVM) to predict emotive and eating aspects [10]. Karuppusamy, et.al. has elaborated the recurrent neural networks with the ability to predict consumer behaviour which enables organizations and marketers to develop successful marketing campaigns and forecasts. Using the regularly occurring essential need for prediction [11].

In “Machine Learning Algorithms - A Review” [14] have stated that there are many algorithms. When we can use lots of data in datasets then for in supervised an unsupervised algorithm in different ways for better results and performances. A

study on “Unsupervised machine learning via transfer learning and k-means clustering to classify materials image data” [15] have used unsupervised learning algorithm that is k-means for Unsupervised machine learning gives better possibilities for improving machine learning performance and for deducing knowledge from unlabelled data sets [16].

3. Methodology

There are three different parts:

- Data collection.
- Data Pre-processing.
- Cluster Creation.

A Python programme has been created, and Jupyter Notebook is used to run the programme after importing the appropriate library. Pandas – Loading & Pre-processing of data, Numpy – Numeric Data Calculation. Scikit - Learn Machine Learning library, IPython-matplotlib – Visualization.

3.1 Data Collection

The primary data used in this study is collected through the survey. Total 320 responses were collected for analysis. All of the responses in this study were collected through randomly selected restaurant customers. This study is based on local family restaurant situated at Kolhapur city, of Maharashtra State. This restaurant was established in year the 1997 and is one of the popular family restaurants in the city.

The survey was carried out over a period of three-month, from June to August 2022. Six important factors are examined in this study that are addressed in the questionnaire: Ambient, Meal quality, Information system, Responsiveness, Empathy, and Assurance. This procedure provides more accurate responses to our questionnaire and aids respondents in understanding the measure in detail.

Following are some sample questions from the survey:

- Which type of food you liked in the restaurant?

- Which service you liked most in the restaurant?
- What you liked the best in the restaurant?
- Are you satisfied with the food quality and service?
- Which type of ordering mode you prefer for ordering food from restaurant?

These some questions were asked in the google form with option of Likert scale format requested in order to get feedback.

We have used a method of data collection using online forms to complete. Throughout, order to choose a number of questions for the clustering prediction in the current investigation. Additionally, data cleaning techniques are used to check for missing values in the data processes. We have used Open Refine for data cleaning. Open Refine is an open-source desktop software for data preparation, which is the process of cleaning up data and transforming it into different formats. It can handle spreadsheet file types like CSV and is similar to spreadsheet programmes. So, Data cleaning steps likes remove duplicate or irrelevant observations, fix structural errors, filter unwanted outliers, handle missing data, and validate and QA.

3.2 Data Pre-processing

In data pre-processing Fig. 1. shows the specific question and its relevant elements. Each user only received one Google form including 20 questions during the data collection. Fig. 1 below that a variety of features that can be used to evaluate the restaurant's quality. The factors in this study were measured using a 4 (four) point scale, with 1 denoting poor, 2 adequate, 3 good, and 4 excellent (excellent). The solutions to questions 6, 7, 8, 10, and 18 were allowed.

The dataset which is an CSV file is imported using Pandas.

	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	...	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20	Q21	Q22	
0	SWAPNIL	VINAYAK	M	Maharashtra	Ichalkaranji	1	4	4	4	3	...	4	B	Hygiene	Food	Visiting	4	4	Parcel	4	Friends/Family	
1	Suraj	Dharanu	M	Karnataka	Nipani	1	3	3	Taste of food,	2	3	...	3	B	Quick service	Food Quality	Online	3	3	Animal bite	3	Self Visiting
2	Kishor	Neha	M	Maharashtra	Kolhapur	1	2	3	Variety of	2	2	...	3	B	Service with a	Famous	Visiting	3	3	Parcel	3	Social Media

Fig. 1: Dataset loaded using Pandas.

The data from the entire document was exported to a CSV file and given some null values. The responses to questions 1 to 20 were transformed into a number ranging from 1 to 4, which represents level of customer satisfaction. Due to the type of answer being "String," many questions were skipped not be addressed for the following step.

```

In [2]: features = ["Q6","Q7","Q8","Q10","Q18"]
In [3]: restaurant = restaurant.dropna(subset=features)
In [4]: data = restaurant[features].copy()
In [5]: data
Out[5]:
   Q6  Q7  Q8  Q10  Q18
0   1   4   4    4    4
1   1   3   3    2    3
2   1   2   3    2    3
3   1   4   4    4    4
4   1   4   4    3    3
...
315  2   2   3    4    3
316  1   3   3    4    3
317  1   4   4    4    4
318  1   4   4    4    4
319  2   4   4    3    3

```

320 rows × 5 columns

Fig. 2: Questions Importing.

We import questions only data as the numeric present such as fig. 1. According to the Likert scale, which has a scale of 1 to 4, poor, adequate, good, and exceptional, the factors in this study were measured. In Figure 2 served as the basis for the questions 6, 7, 8, 10, and 18.

3.3 Cluster Creation

Clustering is an unsupervised learning method that is used to look for natural groupings of data and patterns within the data. The hierarchical technique, the expectation-maximization technique, and the K-Means technique are all examples of clustering techniques. Cluster creation benefits when elements are clustered, productivity increases, decision-making is simplified, and new opportunities are produced.

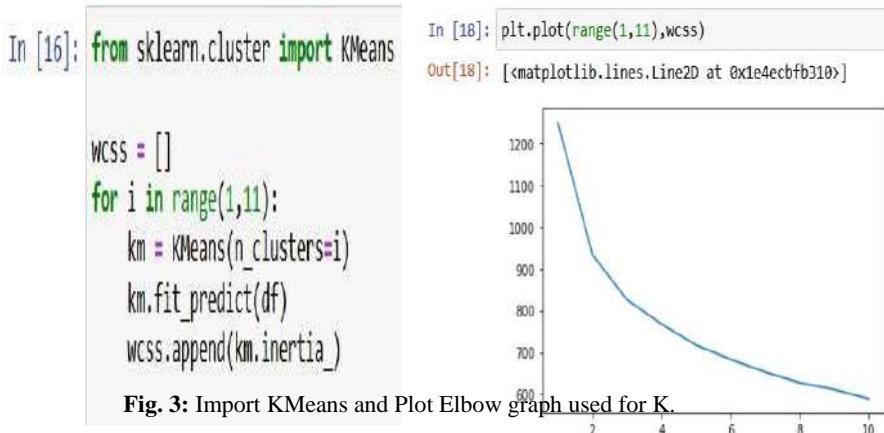


Fig. 3: Import KMeans and Plot Elbow graph used for K.

First, we import KMeans from sklearn.cluster and plot a graph. we use the elbow approach in Fig. 3 above to select K number of clusters. The model we used to analyse our pre-processed data was K-Means Clustering. Also used elbow method to calculate clusters creation. Every clustering result's data visualisation used the Principal Component Analysis (PCA). An unsupervised learning method for lowering the dimensionality of data is principal component analysis. While minimising information loss, it simultaneously improves interpretability. It makes data easier to plot in 2D and 3D and aids in identifying the dataset's most important properties. One of the most used clustering methods is K-Means. due to a suitable remedy that has been put to use for many challenging tasks. Formula (1) illustrates how the K-Means model.

In [32]: restaurant[labels == 0][["Q5"]+features]							
Out[32]:							
		Q5	Q6	Q7	Q8	Q10	Q18
0	Ichalkaranji	1	4	4	4	4	4
3	Paranda	1	4	4	4	4	4
4	Kolhapur	1	4	4	3	3	3
5	Pune	2	4	4	4	4	4
8	Kolhapur	1	4	3	4	3	3
...
308	Satara	1	4	4	4	3	3
312	Osmanabad	1	4	4	4	3	3
317	Osmanabad	1	4	4	4	4	4
318	Osmanabad	1	4	4	4	4	4
319	Pune	2	4	4	3	3	3

208 rows × 6 columns

(A) Label 0.

Lebel 0 cluster 1 has 208 customers who are very happy with services provided by the restaurant. Most of the customers in this cluster have visited the restaurant for first time. Here customers are satisfied with the ambience, seating arrangement and food quality. Lots of customers are interested for ordering food in the restaurant. Customers are satisfied with the hygiene condition of every part in the restaurant.

```
In [33]: restaurant[labels == 1][["Q5"]+features]
```

```
Out[33]:
```

		Q5	Q6	Q7	Q8	Q10	Q18
1	Nipani	1	3	3	2	3	
6	Paranda	1	3	3	3	3	
7	Paranda	1	3	3	4	3	
10	Bhubaneshwar	1	3	3	4	3	
11	Paranda	1	2	3	3	3	
...	
310	Panaji	1	3	2	3	3	
311	Tuljapur	1	1	2	3	4	
314	Tuljapur	1	4	3	3	2	
315	Tuljapur	2	2	3	4	3	
316	Ahmednagar	1	3	3	4	3	

96 rows × 6 columns

(B) Label 1.

Cluster 2 are satisfied with the services. 96 of the customers in Lebel 1. Many of the customers in this cluster are happy with the service's availability for all customers. The majority of customers want to collect their items from the restaurant. However, a huge percentage of customers are new arrivals, while some are regulars. The variety of foods attracts customers' interest.

```
In [34]: restaurant[labels == 2][["Q5"]+features]
```

```
Out[34]:
```

		Q5	Q6	Q7	Q8	Q10	Q18
2	Kolhapur	1	2	3	2	3	
21	Kolhapur	2	2	2	2	3	
26	Ichalkaranji	2	2	3	1	2	
40	Sangli	1	2	2	2	2	
228	Siwan	1	3	3	3	1	
230	delhi	2	3	2	2	3	
233	delhi	1	3	2	3	1	
244	Pune	1	4	2	2	3	
252	delhi	1	3	2	2	3	
264	Pune	1	4	1	2	2	
255	Nagpur	1	2	2	2	2	
260	Dharbhanga	1	2	2	2	2	
268	Kolkata	2	2	2	1	2	
272	Kolkata	2	3	2	1	4	
278	Dharashiv	1	3	2	2	2	
313	Pune	2	3	2	2	3	

(C) Label 2.

Here are 16 customers who were less satisfied with the restaurant were indicated in Lebel 2 of Cluster 3, which is. Only the hygienic condition makes these customers satisfied. These specific customers are simply interested in quality services. The majority of these customers are doing online food orders.

4. Result and Discussion

In the form 320 questionnaires with 20 features and 2 Boolean features were obtained from this survey. 22 questions were answered by the questionnaire's respondent. There are five questions selected for clustering. The most common data were used to fill in the missing values to resolve this issue. The size of the cluster for each methodology is shown in Table 1. We can observe that K-Means splits the data properly equally among each cluster in the case of scale. So, there are three clusters made, identified as clusters 1, 2, and 3.

Table 1: Data contains of Clusters.

Cluster Model	Cluster 1 (Label 0)	Cluster 2 (Label 1)	Cluster 3 (Label 2)	Total
<i>K-Means Clustering</i>	208	96	16	320

Table 1. displays the principal component clustering of the results using K-Means. The results of the data are already successfully clustered visually, as seen by the relatively even distribution of the data in each cluster. The results are consistent with the claim that the K-Means approach's ability to group a large number of data points depends on the number of defined groups.

The Euclidean distance measures the difference between two vectors with real values. When measuring the distance between two rows of data that contain numerical values, such as floating point or integer values, Euclidean distance is most frequently used. It is typical to normalise or standardise the numerical values across all columns if columns contain values with different scales before figuring out the Euclidean distance. Otherwise, the distance measure will be dominated by columns with high values. Therefore, it is calculated Euclidean distance by using formula (1). For the measuring distance of the centroids as given below.

```
distances = centroids.apply(lambda x : np.sqrt(((data - x)**2).sum(axis=1)))
```

(1)

Formula (1): Calculating distance of centroids.

```
In [37]: KMeans(n_clusters=3)
Out[37]: 
+     KMeans
KMeans(n_clusters=3)

In [38]: centroids = kmeans.cluster_centers_
In [39]: pd.DataFrame(centroids, columns=features).T
Out[39]:
   0      1      2
Q6  1.447154  1.421053  1.454545
Q7  3.666667  2.855263  3.793388
Q8  3.666667  2.671053  3.735537
Q10 3.723577  2.828947  3.851240
Q18 2.902439  2.973684  4.000000
```

Fig. 4: Centroids of Clusters.

The elbow method is used to find clusters using centroids. These are the centroids of the features utilised in the data set's K-Means computation of Euclidean distance. Euclidean distance is used in machine learning algorithms as a default distance metric to measure the similarity between two recorded observations. Additionally, each row represents a feature and each column has centroids. Basically, iteration used for better result of cluster creation.

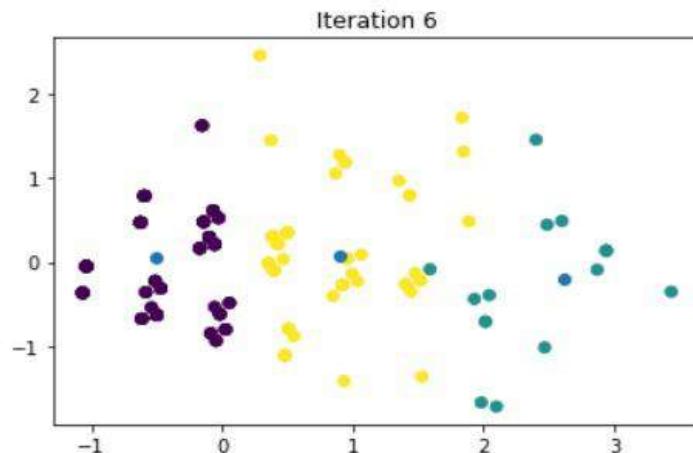


Fig. 5: Result of cluster using K-Means.

Here, three clusters are made of based on the various K clusters using iterations for better result, a cluster-predict approach evaluation could be carried out. For the clustering method shown in fig. 5, we ran our experiment with K = 3.

5. Conclusion

Dataset is created using survey data. A survey was carried out to collect feedback of the restaurant. This restaurant dataset was partitioned into three clusters using K-Means algorithm. The first cluster has all the customers who visited the restaurant for first time and are very happy with the service and food quality. Many of customers are interested in placing food orders at the restaurant. Customers are impressed with the hygiene of every aspect of the restaurant. The second cluster has all the customers who are happy with the service's availability for all customers. The majority of customers want to collect their remaining food items from the restaurant. While some customers are regulars, a substantial percentage of them are newcomers. The good variety of food attracted most customers. The third cluster customers are only satisfied by a hygienic condition. These specific customers are only concerned with getting only good services. The majority of these customers order food in online mode.

So, restaurant can review this clusters and also can design their management of food qualities, services Etc. In order to identify the qualities of restaurant that match customer expectations as well as required improvement. The findings of this study will be helpful in providing advice and suggestions on the hierarchy and priority for the growth and enhancement of restaurants in the future. Additionally, the recommended clustering method can also be improved by collecting more samples and developing a more relevant questionnaire that focuses on a particular area of restaurant marketing strategies.

References

- [1]. Niphat Claypo, Saichon Jaiyen.: Opinion Mining for Thai Restaurant Reviews using K-Means Clustering and MRF Feature Selection, 978-1-4799-6049-1, 2015.
- [2]. Kuppani Sathish, Somula Ramasubbareddy. K.Govinda, E.Swetha.: Restaurant Recommendation System Using Clusteringtechniques, IJRTE, 2019.
- [3]. Shina, Shikha Sharma, Anshu Singla.: A Study of Tree Based Machine Learning Techniques for Restaurant Reviews, ICCCA, 2018.
- [4]. Abdullah Alghamdi.: A Hybrid Method for Customer Segmentation in Saudi Arabia Restaurants Using Clustering, Neural Networks and Optimization Learning Techniques, Arabian Journal for Science and Engineering, 2022.

- [5]. Elham Asani, Hamed Vahdat-Nejad, Javad Sadri.: Restaurant recommender system based on sentiment analysis, Elsevier Ltd, 2021.
- [6]. Akshay Krishna, V. Akhilesh, Animikh Aich and Chetana Hegde.: Sentiment Analysis of Restaurant Reviews Using Machine Learning Techniques, Emerging Research in Electronics, Computer Science and Technology, Lecture Notes in Electrical Engineering 545, 2019.
- [7]. K Purwandari, J Sigalingging, M Fhadli Et al. : Data Mining for Predicting Customer Satisfaction Using Clustering Techniques, ICIMTech, 13-14, 2020.
- [8]. Shina, Shikha Sharma, Anshu Singla.: A Study of Tree Based Machine Learning Techniques for Restaurant Reviews, International Conference on Computing Communication and Automation, 2018.
- [9]. Elham Asani, Hamed Vahdat-Nejad, Javad Sadri.: Restaurant recommender system based on sentiment analysis, Elsevier Ltd., 2021.
- [10].Yi Luo and Xiaowei Xu.: Predicting the Helpfulness of Online Restaurant Reviews Using Different Machine Learning Algorithms: A Case Study of Yelp, Sustainability, 2019.
- [11].Karuppusamy, Dr P. "Artificial Recurrent Neural Network Architecture in Customer Consumption Prediction for Business Development." Journal of Artificial Intelligence and Capsule Networks 2, 2020.
- [12].Aljalbout, Elie, et al. "Clustering with deep learning: Taxonomy and new methods." 2018.
- [13].Károly, Artúr István, Róbert Fullér, and Péter Galambos. "Unsupervised clustering for deep learning: A tutorial survey." *Acta Polytechnica Hungarica* 15.8 2018.
- [14].Mahesh, Batta. "Machine learning algorithms-a review." *International Journal of Science and Research (IJSR). [Internet]* 9 (2020).
- [15].Cohn, Ryan, and Elizabeth Holm. "Unsupervised machine learning via transfer learning and k-means clustering to classify materials image data." *Integrating Materials and Manufacturing Innovation* 10.2 (2021).
- [16].Sinaga, Kristina P., and Miin-Shen Yang. "Unsupervised K-means clustering algorithm." *IEEE access* 8 (2020).

Towards the Implementation of Traffic Engineering in SDN: A Practical Approach

*Prabu.U¹, Geetha.V²

¹Department of Computer Science and Engineering,
Pondicherry Engineering College, Puducherry, India

¹uprabu28@gmail.com

²Department of Information Technology,
Pondicherry Engineering College, Puducherry, India

²vgeetha@ptuniv.edu.in

* Prabu.U

Abstract. The advancement of software-defined networks (SDN) plays a major role in the next generation of networks. It has laid its root in the cloud, data center, and the internet of things. SDN separates the data and control plane. Through this, it provides greater flexibility, cost-effective configuration, and improved functioning of advanced design of networks. Traffic engineering (TE) is one of the major strengths of software-defined networks. Its main goal is to manage the flows in the network, update topologies, analyze the traffic, and be fault tolerant. Apart from that, it performs load balancing, congestion control, traffic rerouting, and updating policies. This article is intended to provide a detailed description of the types of TE techniques, formulation of TE problems, different ways to implement TE in SDN, and different topologies to be considered for the performance evaluation of the algorithms. The overview of the above-mentioned things might be beneficial for beginners and future researchers to have a clear understanding and go ahead with the implementation of TE in SDN.

Keywords: Software Defined Networks, Mininet, Controllers, Linear Programming, Topology

1 Introduction

Software Defined Network (SDN) is a network structure that separates the control function and data forwarding functions. It centralizes intelligence and abstracts the underlying structure from services and applications. It comprises layers such as infrastructure, control, and application. The infrastructure layer consists of either physical or virtual switches. These switches just forward the packets. The control layer acts as an instructor to take all decisions ie., routing algorithm, etc. The application layer comprises various applications for network management, analysis, and other business functions. The SDN architecture is shown in figure 1.

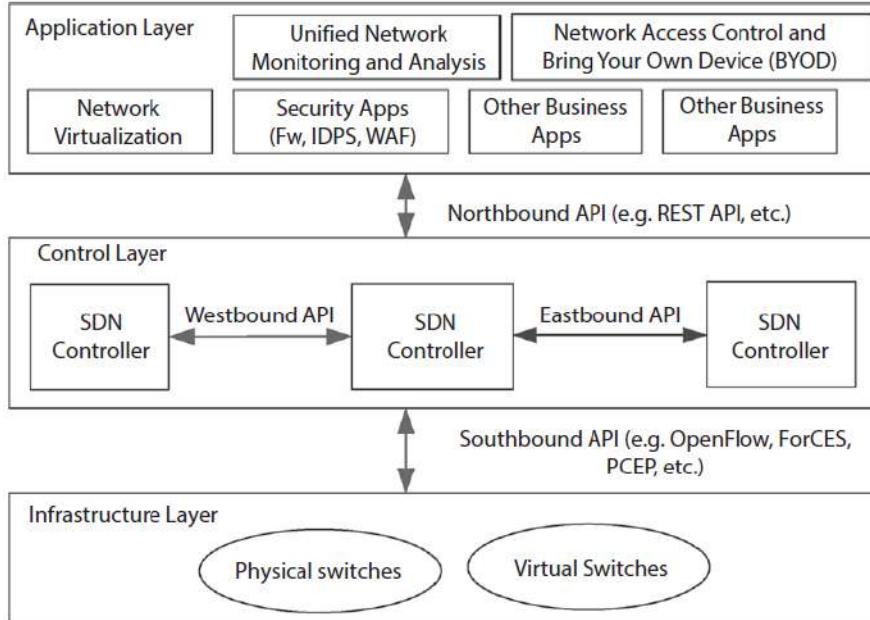


Fig. 1. SDN Architecture [1]

The processing of flow in SDN is given as follows [2]: Whenever the switch receives the initial packet it forwards it to the controller, and it decides the forwarding path. Then the flow tables get installed at every switch. Thereafter all the successive packets get forwarded without any intervention from the controller. During such an operation there occurs considerable overhead due to more number of new flows and even the time taken for framing the forwarding rules. Hence to resolve these challenges traffic engineering techniques should be devised.

The structure of the article is as follows: section 2 details the concepts of TE in SDN, section 3 provides the problem formulation, section 4 describes different ways to implement TE in SDN, section 5 offers topology for performance evaluation, and section 6 concludes the article.

2 Traffic Engineering in SDN

Traffic Engineering (TE) is one of the significant features of SDN. It is employed in the control plane. It is system-oriented and its task is to decide the routing of the traffic and balance the load for the given topology and traffic matrix (TM) with less loss rate and latency. The load balancing enhances the performance of the networks by minimizing congestion, reducing operational costs, and distributing the bandwidth efficiently. The traffic in the network will keep on changing and to deal with such traffic, the TE is classified as static, dynamic, and semi-static [3].

In static, the TMs are observed over a while to reconfigure based on the traffic. Dynamic is used for predicting the consequent state through the previous TMs. Semi-static combines both static and dynamic for optimal configurations. Efficient TE methods are important for utilizing network resources and providing better performance. It also allots the bandwidth and modifies the traffic route to optimize the network.

3 Problem Formulation

Initially, the network may be represented as either a directed or undirected graph that consists of nodes and edges. Then there are different ways to formulate the problem for traffic engineering in SDN. The problem is formulated based on the type of issue addressed. The different ways to formulate the problem are listed based on the literature.

- To minimize the maximum link utilization for accomplishing optimal traffic engineering and to ensure the quality of services such as scalability, packet loss ratio, and latency for multicasting and streaming applications the problem may be formulated as an Integer Linear Programming problem.
- The scenarios involving the optimization of traffic flows may be formulated as a Mixed Integer Linear Programming problem.
- To decide the reconfiguration of the network based on the evolution of traffic the problem may be formulated as a Mixed Integer Nonlinear Programming problem.
- To determine the improved mapping of the switch and controller the problem may be formulated as an Optimization problem.
- Once the addressed issue involves multicast routing, cost, and multimedia applications the problem may be formulated as a Delay Constraint Least Cost problem.

4 Different ways to Implement Traffic Engineering in SDN

Traffic engineering in SDN can be implemented in two ways: Simulation / Emulation and Optimization.

4.1 Mininet Emulator

The most basic way to implement traffic engineering in SDN is via the usage of mininet with custom topologies. It can generate a virtual OpenFlow network with hosts, links, switches, and controllers. The traffic flows are captured through the Wireshark analyzer. Apart from the mininet, there are also other emulators, simulators, and tools available for SDN such as Mininet CE, Mininet-HiFi, OMNeT++, NS-3, EstiNet 8.0, and SDN Cloud.

4.2 Controllers

A controller is a program that performs the logic and directs the data plane. There are various controllers available. Before going ahead with the usage of controllers various things to be considered such as the learning curve, community support, programming language (as it may affect the performance), and policy layer for APIs such as southbound API, and northbound API [4].

NOX

NOX [5] was the first generation controller of OpenFlow which was open-source and widely used. It has two flavors namely NOX-classic and the new NOX. NOX-classic came with C++, python as the programming language, and later the support was stopped. The new NOX has only C++ as the programming language. It had a clean codebase and was fast. It was used when the requirements are of low-level facilities and has the semantics of OpenFlow.

POX

The python version of NOX is called POX [6]. It supported only OpenFlow V1.0. It was widely used, well-maintained, and had good support. But it lacks in performance. It may be used for experimentation and rapid prototyping.

RYU

RYU [7] is an open-source python controller. The name RYU is originated from the Japanese word which means flow. It supports OpenFlow versions 1.0 to 1.4. It has well-defined application programming interfaces. It is very useful for OpenStack integration. It supports the extensions of nicira.

Floodlight

Floodlight [8] is an open-source Java controller which supports OpenFlow version 1.0. It is maintained by Big Switch Networks. It can be integrated with REST API. It can be used at the production level and for multi-tenant clouds.

OpenDaylight

OpenDaylight [9] is also a Java controller which is robust and has an extendable codebase. It can be integrated with OpenStack and cloud applications. But it is complex to use. Maestro is a Java-based controller which supports OpenFlow version 1.0. It has a web-based interface and is widely used for research purposes.

Open MUL

Open MUL [10] is a C language-based controller and provides a multithreaded environment. It supports up to OpenFlow version 1.4. It also provides a web-based interface. It can be used for data center applications. It is good for reliability and performance-based scenarios.

Trema

Trema [11] is C and Ruby-based controller. It supports OpenFlow versions 1.0 and 1.3. It is used for research purposes. It provides rich library support.

Beacon

Beacon [12] is Java based controller. It supports OpenFlow version 1.0. It provides a web-based interface. Iris is Java based controller which supports OpenFlow versions 1.0 and 1.3.

Rosemary

Rosemary [13] is a C language-based controller. It supports OpenFlow versions 1.0 and 1.3. It can be used for research purposes within the campus. ParaFlow is a C++-based controller.

All the aforementioned controllers are centralized. There are also many distributed SDN controllers namely ONOS, Onix, B4, Kandoo, Espresso, Ravana, ODL, DISCO, Hydra, IRIS-HiSA, ElastiCon, and Orion.

Another way of implementation is by considering the network as a graph and framing the problem as a linear programming problem. In this case, the problems are solved by using GNU Linear Programming Kit (GLPK) [14] and optimization software such as the Gurobi solver [15] and CPLEX optimizer [16].

5 Topology for Performance Evaluation

The below table 1 provides the real and standard network topology considered for evaluation of the algorithms in the literature which are part of internet zoo topology [17] and others.

Table 1. Network Topologies

Network	Nodes	Links
18-node EON	18	33
20-node ring	20	20
AARNET	19	24
Abilene	11/11/12	28/14/30
Abovenet	15/17/23	60/74/31
ARPANET	24	100
AT&T	166	189
atlanta	15	22
BellSouth	50	264
Bestel	80	404
BHVAC	19	46
Biznetworks	29	33
Brite10	10	40

Brite15	15	90
Brite20	20	160
BTNA	36	76
Cernet	14/41	32/59
Claranet	15	18
Cogent	196	980
Columbus	70	85
di-yuan	11	42
Ebone	18/28	66/66
EON	19	74
Exodus	21/22	72/51
Geant	23/38	74/104
Geant-2	34	54
German backbone network	14/17	21/26
germany50	50	88
Internet2	10	13
IRIS	51	64
METRO	11	84
nobel-germany	17	26
nobel-us	14	21
NSF	8	20
PACBELL	15	42
pdh	11	34
polska	12	18
Sprint	11/27/30	18/126/138
SURFNET	50	73
ta2	65	108
Tiscali	28	132
USNET	24	43
VNSL	9	22

6 Conclusion

SDN has revolutionized the design of the network. It has also acquired extraordinary attention from industry and research firms. The forthcoming network architectures are dependent on the SDN. Due to the evolving traffic, TE requires a major concentration. So based on this perspective, an in-depth analysis has been done on the implementation strategies of TE in SDN right from the problem formulation to the selection of topology for performance evaluation of the algorithms.

References

1. Fundation, Open Networking. "Software-defined networking: The new norm for networks." ONF White Paper 2.2-6 (2012): 11.
2. Akyildiz, Ian F., et al. "A roadmap for traffic engineering in SDN-OpenFlow networks." Computer Networks 71 (2014): 1-30.
3. Sanvito, Davide, et al. "Clustered robust routing for traffic engineering in software-defined networks." Computer Communications 144 (2019): 175-187.
4. Ahmad, Suhail, and Ajaz Hussain Mir. "Scalability, consistency, reliability and security in SDN controllers: a survey of diverse SDN controllers." Journal of Network and Systems Management 29.1 (2021): 1-59.
5. Gude, Natasha, et al. "NOX: towards an operating system for networks." ACM SIGCOMM computer communication review 38.3 (2008): 105-110.
6. POX. <https://noxrepo.github.io/pox-doc/html/>
7. RYU. <https://ryu-sdn.org/>
8. Floodlight. <https://github.com/floodlight/loxigen>
9. OpenDaylight. <https://www.opendaylight.org/>
10. Open MUL. <https://www.openmul.org/>
11. Trema. <https://trema.github.io/trema/>
12. Erickson, David. "The beacon openflow controller." Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. 2013.
13. Shin, Seungwon, et al. "Rosemary: A robust, secure, and high-performance network operating system." Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. 2014.
14. GLPK. <https://www.gnu.org/software/glpk/>
15. Gurobi Solver. <https://www.gurobi.com/>
16. CPLEX Optimizer. <http://ibmdecisionoptimization.github.io/docplex-doc/mp.html>
17. Knight, Simon, et al. "The internet topology zoo." IEEE Journal on Selected Areas in Communications 29.9 (2011): 1765-1775.

Real time intrusion detection in connected autonomous vehicles

Anjanee Kumar¹ and Tanmoy Kanti Das²

¹ Department of Computer Applications, National Institute of Technology Raipur,
G.E. Road, Raipur 492010, Chhattisgarh, India
akumar.phd2021.mca@nitrr.ac.in

² Department of Computer Applications, National Institute of Technology Raipur,
G.E. Road, Raipur 492010, Chhattisgarh, India
tkdas.mca@nitrr.ac.in

Abstract. Connected Autonomous Vehicles (a.k.a CAVs) have shown a new paradigm in the transportation field by reducing cost, managing traffic, and efficiently use of fuels. These developments have revolutionised not only the transportation field but also impacted our daily lives. However, growth in the field of automation has given rise to various security issues. Now CAVs are using many sensors to perform automation. For proper navigation of vehicles, these sensor informations must be communicated in a safe environment. In CAVs, these sensor informations communicates through In-Vehicle-Network (IVN) to various parts of the vehicle, including different ECUs. But the Network environment that has been used for communication is not always safe and can be infiltrated easily if proper security measures have not been taken. We are proposing a new Intrusion Detection System (IDS) to detect intrusions in real-time in the network field of CAVs that is based on Logical Analysis of Data (a.k.a LAD). For this, we have used the CAV-KDD dataset, which has been developed from a benchmark dataset KDDCUP'99 and resembles an actual network environment of CAVs. Our results has shown better performance than the existing results on CAV-KDD dataset.

Keywords: Logical Analysis of Data (LAD) · Intrusion Detection System · Connected Autonomous Vehicles · security.

1 Introduction

From purely mechanical devices, the CAVs have emerged as cyber-physical devices with many connected components. There are multiple processors embedded in the CAVs which communicate over the network in the car, known as In-Vehicle-Network (IVN). The different components in CAVs are connected to the outside world via a wire or wireless network. There are different ways of communication in CAVs, like intra- communication, which includes communication between different components within the vehicles and inter-communication, which includes communication with other CAVs. There are various components in CAVs, including sensors, actuators, and different Electronic Control Units

(ECUs), which are connected to communicate over IVN. Most of these IVNs have securities issues like a lack of authentication and encryption. One such IVN is Controller Area Network (CAN), which is the de facto standard for in-vehicle communication in CAVs. The automotive industry's evolution has also given rise to cyber-attacks as vehicles were exposed to the external world via different network environments. There have been a number of successful attacks against CAVs, which have had devastating effects on public safety. As the automotive industry has been reached every corner of the world and growing continually at an exponential rate, they have become a most likely target for attackers. In 2015, researchers found the vulnerability in the vehicle by hijacking it. This remote exploitation of cars was demonstrated in a seminal paper [13], where the white hat hackers successfully took control of the Jeep Cherokee and exploited it. It has been found in a study that the automobile industry could lose approximately 1.1 billion for a single attack. And before 2023, the entire industry is estimated to lose up to 24 billion. These numbers are scary if proper security measures have not been adopted.

Although there are many methods to deal with cyber-attacks in the system, an intrusion Detection System (IDS) is one such method that a number of researchers have used. As described in [11], IDS detects unauthorized intruders and attacks in the network (In-Vehicle-network in case of CAVs). There are two categories of IDS as mentioned in [11]: online IDS, where the detection and analysis of traffic are performed in real-time and offline IDS, where the detection and analysis of logs are performed sometime after the network operation. Here in this paper, we have focussed on a real-time intrusion detection system. As CAVs are safety critical Cyber-Physical-System where a delay in detecting attacks may have a devastating effect on public safety. One of the most important points while considering the IDS system for CAVs is that the commercial vehicle has limited computation capability. Therefore, most of the recent methods are incompatible with the current scenario. For example, methods involving Deep Learning has excellent detection result [8][17]. Still, it requires a huge amount of processing power and high computing resources, which may not be available in the CAVs. Therefore, we are proposing a real-time (online) intrusion detection method based on Logical Analysis of Data.

LAD is a binary classifier method proposed by Hammer et al. [2] to find the behavioural patterns of normal and abnormal activities in a system. These abnormal activities may include any cyber-attacks to discover any anomalies or intrusions. The process of extracting patterns requires significant knowledge of the domain. These patterns must differentiate the normal activity from the abnormal activity even in any sensor noise in a system network. In general, LAD is a method to generate such patterns from historical observations. In LAD, a historical observation dataset \mathcal{D} consists of two disjoint datasets $\mathcal{D}+$ and $\mathcal{D}-$. They represent two different classes, the positive class and the negative class. In LAD, we mainly extract prime patterns which can classify any given new observation as either a member of $\mathcal{D}+$ or $\mathcal{D}-$ class. In this paper, we have developed a real-time intrusion detection system based on the LAD method for network sys-

tems in CAVs with optimal computing efficiency. For our proposed method, we have chosen CAV-KDD dataset [7], which is generated from a well-known KD-DCUP99 dataset. This dataset is chosen because it resembles a general network environment which has all possible attacks of CAVs. The main contributions of this paper are as follows.

- **Computationally efficient algorithm:** As mentioned, Vehicles may not have the computational resources; therefore, we have proposed a computationally efficient method that can be achieved by using laptop-class processing power.
- **Real-time IDS:** We have proposed an online real-time IDS which can detect intrusions in real-time. The intrusions in safety-critical systems like CAVs must be detected in real-time to avoid any accidents. The detection time in our method shows that our system is a real-time IDS.
- The **effectiveness** of the proposed method is analysed using different performance metrics, including accuracy, False-positive Rate, F1-Score and detection time.

The paper is organised as follows: In section 2, we have presented related work. In section 3, there is a complete description of dataset. A complete description of our proposed method is presented in section 4. Then the experiment and results were discussed in section 5. Finally, we have concluded the paper in section 6.

2 Related Work

The study in the field of IDS in CAVs is quite vast, and many researchers have contributed to this field by adopting different approaches [19] [9]. Some of them have adopted the concept of time arrival of CAN-BUS messages. In [16], the researchers have analysed each CAN-BUS message and measured the time of arrival of the CAN-BUS message to detect a DDoS attack. They have achieved accuracy up to 100% for DDoS attacks. In comparison to spoofing attacks, DDoS is quite easy to detect. In [14], the researcher has analysed that the distribution of the inter-arrival times has discriminating power in detecting attacks. Some authors have also adopted the sliding window method to CAVs IDS. In [15], the authors have compared the Shannon entropy-based method to similarity-based IDS and concluded that the similarity-based IDS has low computation complexity. In [12], the authors have detected replay and fuzzy attacks by using Shannon entropy. They have adopted a sliding window approach and also addressed the spoofing attacks. In [10], authors have also used the time interval of the CAN-BUS messages to detect spoofing attacks. They have used the time interval information and the correlation coefficient between offsets to detect various attacks.

In [18], to determine the presence of attacks, the authors have evaluated the impact of different window sizes. They have also used the CAN-ID feature. Finally, they have shown that spoofing attacks are hard to detect. In [6], the authors have applied an improved isolation forest method with data mass for

the detection of attacks, whereas for the isolation forest algorithm, data mass is used as a base function. Also, in [17], a fixed sliding window has been used to transform the in-vehicle traffic to images which are then passed as input in Convolutional Neural Network (CNN) for classification. Finally, the paper [7] that have results on the same dataset that we have used. In this paper, the authors have used Unified Modelling Language (UML) framework to generate the CAV-KDD dataset from a well-known KDDCUP'99 dataset. And then, they applied Naïve Bayes and the Decision tree algorithm to calculate different performance metrics.

3 Description of Dataset

3.1 CAV-KDD Dataset

For the generation of CAV-KDD dataset the authors in the paper [7] has adapted KDD99 dataset on intrusion detection system and created a CAV communication-based cyber-attack dataset which is termed as CAV-KDD dataset. They have used UML-based CAV framework which is explained in [7] to find out the possible attacks in CAVs system. Out of total 39 sub-attacks present in KDD99 dataset, they have selected 14 sub-attacks that is possible in CAVs system. For preparation of CAV-KDD dataset, they have used 10% of the KDD99 training and testing dataset and carved out CAV-KDD dataset based on the discussed UML framework of CAVs system. The authors in the paper [7] have provided the number of records of CAV-KDD test and CAV-KDD train dataset that we have used in this paper. All the explanation of CAV-KDD dataset can be found in paper [7]. The details of the CAV-KDD training and testing dataset are given in Table 1.

Table 1. Description of CAV-KDD dataset

Dataset	Number of		
	Attacks	Normal	Total
CAV-KDD Training Data	13284	58716	72000
CAV-KDD Testing Data	23349	47913	71262

The description of Sub-attacks in the CAV-KDD Dataset is shown in Table 2.

3.2 Data Preprocessing

The method of LAD can be only applied to the numerical data. Therefore we have preprocessed the dataset to use in our proposed method. The following steps have been taken in order to preprocess the dataset:

Table 2. Description of sub attacks in CAV-KDD dataset

Attack Type	Sub Attack	Training dataset	Testing dataset
PROBE	0 NORMAL	58716	49713
	1 ipsweep	341	455
	2 nmap	158	80
DoS	3 mailbomb		308
	4 neptune	12281	20332
	5 pod	40	45
	6 smurf	199	936
	7 teardrop	199	12
	8 udpstorm		2
	9 buffer overflow	—	22
	10 httpunnel	—	146
	11 ftp write	8	3
U2R	12 Guess passwd	53	1302
	13 worm	—	2
	14 xnoop	—	4

- Conversion to numerical form: There are total 39 input features and one output feature in the CAV-KDD dataset as given in paper [7]. Out of 39 features there are three categorical features namely '*Protocol_Type*', '*Service*' and '*Flag*'. We have converted these categorical features into numerical form using one-hot-encoding method. After conversion we have got 115 features.
- In the output column, we have converted all attacks into anomaly as our proposed method works on binary classification. Finally, in the output column, we have only two entries, i.e., '*anomaly*' and '*normal*'. We have changed '*anomaly*' to '1', and '*normal*' to '0' to finally convert the dataset into numerical form.

4 Introduction to LAD

Logical analysis of data (a.k.a LAD) is one of the effective Boolean based method which is based on *Boolean logic* and *discrete optimization* [3]. The LAD techniques were first developed for binary data [4], and later, it was extended for non-binary data. LAD is a method which finds out the minimal features from the dataset, which is capable of explaining all the observations and can classify the observation into negative and positive observations. The positive and negative observation is based on the classes to which the observation belongs. To apply the method of LAD, all the input variables (explanatory variables) must be converted into binary by means of a discretization process called *binarization*. The binarized dataset \mathcal{D} contains two subsets $\mathcal{D}+$ representing a positive class of data and $\mathcal{D}-$, which represent a negative class of the dataset. LAD extracts meaningful patterns(rules) from binary dataset \mathcal{D} . The binarized data \mathcal{D} can be

expressed as a Boolean function f as $f: \{0, 1\}^n \rightarrow \{0, 1\}$. LAD uses the concept of *partially defined Boolean function*. The observations belonging to the positive class (negative class), i.e., $\mathcal{D}+$ ($\mathcal{D}-$), are termed as True Points (False Points) and denoted as T (F). The dataset \mathcal{D} ($\mathcal{D} = \mathcal{D}+ \cup \mathcal{D}-$) can be represented as *partially defined Boolean function* (pdBf) ϕ as $\phi: \mathcal{D} \rightarrow \{0, 1\}$. The main objective of the LAD method is to find out the extension \mathcal{T} of pdBf ϕ such that it can classify any new observation. Generally, the LAD functions in 5 stages, including:

1. Binarization of dataset
2. Support set generation
3. Pattern generation
4. Theory Formation
5. Classifier Design

Although, we have not applied the *Theory Formation* step as explained in the advanced version of LAD in [5].

4.1 Binarization of dataset:

LAD method begins with the process of *binarization* where each column c_i is converted to binary attributes with the help of cut points. The value n is converted into binary using a threshold value (cut-points) h as if n is greater than h ; it is converted to 1 else 0.

The numerical value n with threshold value h_i is converted to binary and is represented by:

$$\zeta(n, h) = \begin{cases} 1, & \text{if } n \geq h_i \\ 0, & \text{otherwise} \end{cases}$$

We can find out the cut-points of a particular column to convert numerical values to binary form with the following steps:

1. Sort the data in ascending order, including the Class column.
2. Ignore all the observation if it has the same value and keeps only one with a Class label value other than 1 and 0.
3. Repeat step 2 until all observations are distinct.
4. Cut-point(h) can be calculated as $h_j = \frac{1}{2}(n_i + n_{i+1})$

4.2 Support set generation:

In the second step, after getting the binarized data, we need to remove redundant features from the dataset. The process of selecting minimal features that can classify the observation into different classes, such as the intersection of these two classes, gives null value as $\mathcal{D}+ \cap \mathcal{D}- = \emptyset$ is termed as *support set generation*. The set of minimal features is known as the support set. To select this set of minimal features, we have used the “Mutual Information Greedy” algorithm [1]. The elements in our support set are arranged according to their *discriminating power*.

4.3 Pattern generation:

The third step is where we generate meaningful patterns to classify the maximum observation as possible. We have generated these meaningful patterns with the help of binary variables obtained in *support set generation* step. The term \mathcal{M} is defined as the conjunction of literals and is said to cover any observation if $\mathcal{M}(b) = 1$. The binary variables from the support set are written as a term and are said to be a pattern if the conjunction of these binary variables covers observations in the given dataset. If a pattern covers only positive observation (negative observation) but not any negative observation (positive observation) is termed as *pure positive pattern* (*pure negative pattern*). And if this pattern is minimal, i.e., the conjunction of a minimal number of literals, then it is called *prime pattern*. In our implementation, we have also considered a positive impure pattern, which is defined as a pattern which covers maximum positive observations but very few negative observations.

4.4 Classifier design

In our approach, we have skipped the *Theory formation* step as explained in [5]. Based on the set of patterns that we have obtained from the pattern generation step, we have defined a rule-based-classifier, using '*if-else if*' statement. In our implementation, we have grouped all positive pure and positive impure patterns using '*if-else if*' statement to make a classifier.

4.5 Workflow

For our model, we have pre-processed our historical dataset, which involves the conversion of categorical to numerical form. Also, we are working for the binary classifier, so we have considered only two classes in a class label, i.e., *normal* and *anomaly*. All the attacks are put into the anomaly class. The processed dataset is then applied to create a rule-based classifier following the steps as depicted in figure 1. All the steps are well explained in *Algorithm 1*.

Algorithm 1 Steps to design a rule-based Classifier

Input: Historical dataset \mathcal{D} .

Output: Rule-based classifier.

- 1: Convert the categorical column of the historical dataset \mathcal{D} into numerical form.
 - 2: *Binarize* the processed dataset obtained in step 1 using the process explained in section 4.1.
 - 3: Generate the minimal set of binary attributes using the process explained in section 4.2.
 - 4: Extract *positive pattern* using the support set obtained in step 3.
 - 5: Design *Classifier* from the extracted patterns.
-

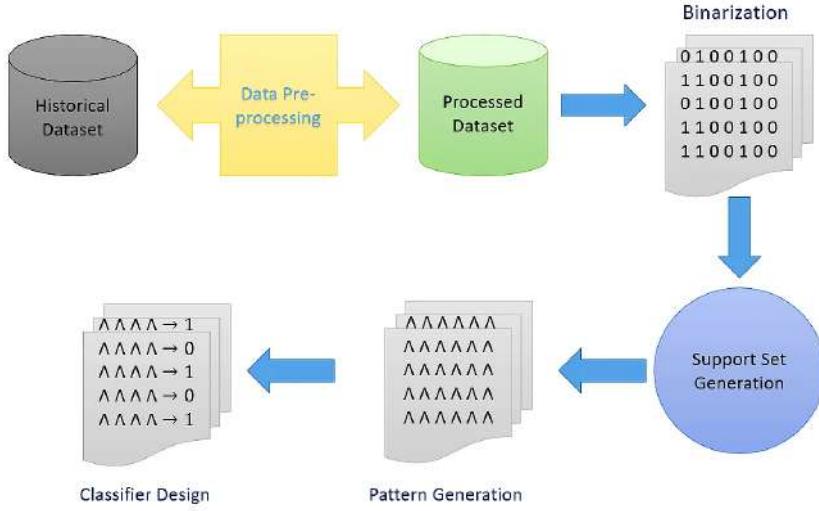


Fig. 1. Workflow of proposed method

5 Performance Evaluations

The LAD-based Intrusion Detection System for the connected autonomous vehicle has been evaluated on the CAV-KDD dataset [7]. We have performed our experiments on a PC with a core i5 processor and RAM of 20GB.

5.1 Performance Metrics

To validate our proposed model, we have calculated and compared the results using following metrics:

The meaning of acronym used in definition of metrics:

- TP (*True Positive*): The examples which are actually positive and predicted positive.
- TN (*True Negative*): The examples which are actually negative and also predicted negative.
- FP (*False Positive*): The examples which are predicted positive but actually negative.
- FN (*False Negative*): The examples which are actually positive but predicted negative.

1. *Accuracy*: It is defined as ratio of correctly classified examples to the total number of examples.

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

2. *Precision*: Precision is defined as fraction of truly positive data out of total predicted positive data and is given by:

$$\text{Precision} = \frac{(TP)}{(TP + FP)}$$

3. *F1 Score*: F1_score shows the balanced value of precision and recall and defined as harmonic mean of recall and precision. It is defined as:

$$\text{F1 Score} = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

4. *False positive rate*: It is defined as ratio of false positive data to the actual negative data and is given by

$$\text{False Positive rate} = \frac{(FP)}{(FP + TN)}$$

5.2 Experimental setup and results

There is a total of 41 explanatory variables and one dependent variable, i.e., attack label, in both the training and testing dataset. In the pre-processing stage, all the categorical columns have been converted to numerical form, and the attack label is labelled as 1 for attack and 0 for normal. In our experiment, we considered attack as a positive class and normal as a negative class. Then we applied LAD to the training dataset and followed all the steps described in Algorithm 1. The following results have been summarised step by step.

1. *Binarization* : In this step, we have got a total of 1491 binary variables.
2. *Support Set Generation* : In the Support set, we have got a total of 25 binary variables.
3. *Pattern Generation* : In this stage, we have got a total of 10 positive patterns, out of which 8 were pure patterns, and two of them were impure.
4. *Classifier Design* : Based on the patterns generated from the previous step, we have designed a rule-based IDS using "if – else – if" statement. The details of the rules have been summarised in defined Classifier 1 .

Based on the rule-based classifier, we have calculated different performance metrics on the training and testing dataset. The results have been summarised in Table 3 and Table 4.

On applying the obtained classifier on the testing dataset, we have got the *accuracy* of 97.34% and *precision* of 99.75%. The detection time of our rule-based classifier is only 31.15 microsecond. The *F1_Score* and the *False_Positive_Rate* obtained are 95.79% and 0.1%. All the results have been tabulated in Table 3 and 4.

Classifier 1: Classifier to distinguish attack and normal

```

1: if row['wrong_fragment'] >= 0.5 then
2:   'attack'
3: else if row['count'] >= 48.5 and row['flag_SF'] <= 0.5 then
4:   'attack'
5: else if row['count'] >= 48.5 and row['service_ecr_i'] >= 0.5 then
6:   'attack'
7: else if row['flag_SF'] <= 0.5 and row['service_private'] >= 0.5 then
8:   'attack'
9: else if row['srv_count'] >= 3.5 and row['service_eco_i'] >= 0.5 then
10:   'attack'
11: else if row['dst_host_srv_count'] >=
    153.5 and row['dst_host_srv_diff_host_rate'] >=
    0.07500000000000001 and row['protocol_type_icmp'] >= 0.5 then
12:   'attack'
13: else if row['dst_host_srv_diff_host_rate'] >=
    0.07500000000000001 and row['protocol_type_icmp'] >=
    0.5 and row['dst_host_count'] <= 2.5 then
14:   'attack'
15: else if row['dst_host_diff_srv_rate'] <= 0.015 and row['hot'] >=
    0.5 and row['logged_in'] <= 0.5 then
16:   'attack'
17: else if row['flag_SF'] <= 0.5 and row['srv_count'] >=
    3.5 and row['dst_host_srv_count'] <= 19.5 then
18:   'attack'
19: else if row['dst_host_srv_diff_host_rate'] >=
    0.07500000000000001 and row['protocol_type_icmp'] >=
    0.5 and row['dst_host_count'] <= 4.5 then
20:   'attack'
21: else
22:   'normal'

```

Table 3. Accuracy and Detection time in CAV-KDD dataset

Dataset	Accuracy	Detection Time
Training dataset	99.90%	34.86 μ s
Testing dataset	97.34%	31.15 μ s

Table 4. Precision, F1_Score and FPR on CAV-KDD dataset

Dataset	Precision	F1_Score	False Positive Rate
Training dataset	99.98%	99.73%	0.003%
Testing dataset	99.75%	95.79%	0.1%

5.3 Comaprison with existing result

The existing result from the paper [7] was based on *Decision Tree* and *Naive Bayes*. In the paper, they have achieved an accuracy of 97.14% which is less than our obtained result as stated in Table 3. We have achieved a precision of about 99.75% which is far better than the existing result of 94.84%. In terms of the False positive Rate, our model has performed very well by achieving 0.1% as compared to the existing result, which is 5.6%. The detection time of our model is only $31.15 \mu\text{s}$, which is very much more efficient than the existing result of 0.94 seconds. All the existing results and the obtained result are tabulated in Table 5.

Table 5. Comparison with the existing literature on CAV-KDD dataset

Reference	Method	Accuracy	Detection Time	Precision	F1_Score	False Positive Rate
This study	LAD	97.34%	$31.15 \mu\text{s}$	99.75%	95.79%	0.1%
[7]	Decision_tree	97.04%	0.94s	94.64%	NA	5.6%
[7]	Naive_Bayes	95.66%	2.42s	94.84%	NA	5.2%

6 Conclusion

The presented work mainly aims to find out the intrusion in autonomous vehicles as the network system in an autonomous vehicle is not much secure. To prevent cyber-physical systems like autonomous vehicles needs continuous research as any loophole in the security may be devastating to public safety. The attacks need to be detected in real time so as to have fewer consequences of it. Our model, which is based on LAD that involves binarization and selection of important features, provides a real-time detection approach. The lower false positive rate obtained in the dataset suggests that the LAD method is one of the efficient methods of detecting intrusions. Autonomous vehicles involve multiple network systems, and our method is effective and low-cost in case of identifying attacks in any network.

References

1. Almuallim, H., Dietterich, T.G.: Learning boolean concepts in the presence of many irrelevant features. *Artificial intelligence* **69**(1-2), 279–305 (1994)
2. Boros, E., Hammer, P.L., Ibaraki, T., Kogan, A., Mayoraz, E., Muchnik, I.: An implementation of logical analysis of data. *IEEE Transactions on knowledge and Data Engineering* **12**(2), 292–306 (2000)

3. Bruni, R., Bianchi, G., Dolente, C., Leporelli, C.: Logical analysis of data as a tool for the analysis of probabilistic discrete choice behavior. *Computers & Operations Research* **106**, 191–201 (2019)
4. Crama, Y., Hammer, P.L., Ibaraki, T.: Cause-effect relationships and partially defined boolean functions. *Annals of Operations Research* **16**(1), 299–325 (1988)
5. Das, T.K., Adepu, S., Zhou, J.: Anomaly detection in industrial control systems using logical analysis of data. *Computers & Security* **96**, 101935 (2020)
6. Duan, X., Yan, H., Tian, D., Zhou, J., Su, J., Hao, W.: In-vehicle can bus tampering attacks detection for connected and autonomous vehicles using an improved isolation forest method. *IEEE Transactions on Intelligent Transportation Systems* (2021)
7. He, Q., Meng, X., Qu, R., Xi, R.: Machine learning-based detection for cyber security attacks on connected and autonomous vehicles. *Mathematics* **8**(8), 1311 (2020)
8. Kang, M.J., Kang, J.W.: Intrusion detection system using deep neural network for in-vehicle network security. *PloS one* **11**(6), e0155781 (2016)
9. Karopoulos, G., Kambourakis, G., Chatzoglou, E., Hernández-Ramos, J.L., Kouliaridis, V.: Demystifying in-vehicle intrusion detection systems: A survey of surveys and a meta-taxonomy. *Electronics* **11**(7), 1072 (2022)
10. Lee, H., Jeong, S.H., Kim, H.K.: Otids: A novel intrusion detection system for in-vehicle network by using remote frame. In: 2017 15th Annual Conference on Privacy, Security and Trust (PST). pp. 57–5709. IEEE (2017)
11. Lunt, T.F.: A survey of intrusion detection techniques. *Computers & Security* **12**(4), 405–418 (1993)
12. Marchetti, M., Stabili, D., Guido, A., Colajanni, M.: Evaluation of anomaly detection for in-vehicle networks through information-theoretic algorithms. In: 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI). pp. 1–6. IEEE (2016)
13. Miller, C., Valasek, C.: Remote exploitation of an unaltered passenger vehicle. *Black Hat USA 2015(S 91)* (2015)
14. Moore, M.R., Bridges, R.A., Combs, F.L., Starr, M.S., Prowell, S.J.: Modeling inter-signal arrival times for accurate detection of can bus signal injection attacks: a data-driven approach to in-vehicle intrusion detection. In: Proceedings of the 12th Annual Conference on Cyber and Information Security Research. pp. 1–4 (2017)
15. Ohira, S., Desta, A.K., Arai, I., Inoue, H., Fujikawa, K.: Normal and malicious sliding windows similarity analysis method for fast and accurate ids against dos attacks on in-vehicle networks. *IEEE Access* **8**, 42422–42435 (2020)
16. Song, H.M., Kim, H.R., Kim, H.K.: Intrusion detection system based on the analysis of time intervals of can messages for in-vehicle network. In: 2016 international conference on information networking (ICOIN). pp. 63–68. IEEE (2016)
17. Song, H.M., Woo, J., Kim, H.K.: In-vehicle network intrusion detection using deep convolutional neural network. *Vehicular Communications* **21**, 100198 (2020)
18. Wu, W., Huang, Y., Kurachi, R., Zeng, G., Xie, G., Li, R., Li, K.: Sliding window optimized information entropy analysis method for intrusion detection on in-vehicle networks. *IEEE Access* **6**, 45233–45245 (2018)
19. Young, C., Zambreno, J., Olufowobi, H., Bloom, G.: Survey of automotive controller area network intrusion detection systems. *IEEE Design & Test* **36**(6), 48–55 (2019)

Physical Architecture of Linear Feedback Shift Register using Clock Tree Synthesis for Cyber-Physical System

* B. Muthu Nisha¹, V.Nithya², J.Selvakumar³

^{1,2,3}Department of Electronics and Communication Engineering, SRMIST,
Kattankulathur

¹*mb1850@srmist.edu.in

²nithyav@srmist.edu.in

³selvakuj@srmist.edu.in

Abstract. Every Cyber-Physical System must have a linear feedback shift register to produce keys for every encryption method. The physical design of a shift register operating at 500 MHz with linear feedback was examined in this work. Three design aspects from this physical architecture are thoroughly investigated for usage in application-specific integrated circuits. The worst negative slack timing path between registers was studied for clock frequency consumption performance to comprehend the significance of clock tree synthesis. This system transmits the synchronous signal to each succeeding cell via a clock tree. The needed specification is appropriate for a Random Number Generator with Multibit LFSR, Physically Unclonable Function, and Fibonacci and Galois Number Theory in Cyber-Physical Systems.

Keywords: Cyber-Physical System, Worst Negative Slack, Clock Tree Synthesis, Random Number Generator, Physically Unclonable Function

1 Introduction

Physical Architecture is the process of turning a design into physical shapes. Some of the phases include floor layout, placement, clock tree synthesis, and routing [1]. The construction of a netlist from the Register Transfer Level (RTL) is the first stage in physical design. The netlist comprises specifications of a circuit's parts and connections. Floor planning is the essential initial action. Choosing which structures to place adjacent to one another while taking into account space limitations, travel distances, and other component-related constraints are entailed [2].

When assembling each component or block on the die, timing and interconnect length are taken into account. Clock tree synthesis reduces skew by evenly sharing the clock throughout a system's succeeding pieces by adding buffers or inverters [3]. Clock Tree Synthesis (CTS) is the method by which buffers and inverters along the clock routes of the Application Specific Integrated Circuit (ASIC) architecture synchronize the clock delay to all clock inputs [4]. CTS is therefore deployed to balance the skew and lessen insertion latency.

Worst-case Negative slack is the parameter used to analyze the most negative of any single slack of the timing paths that failed at any design constraint. The duration of time expected for the intake to a flip-flop to be stable before a clock edge is termed setup time. In contradiction to setup time, hold time addresses events that occur after a clock edge. The hold time is the shortest period necessary for the intake to a flip-flop to stay steady following a clock edge [5].

To maintain and monitor the physical environment, cyber-physical systems (CPS) typically incorporate embedded computers with sensor connectivity. Encryption must remain a part of this system for data security in that particular case. To generate keys for every encryption technique, each cyber-physical system must have a linear feedback shift register[6].

This study takes into account setup and holds time analysis on the chosen design's worst-case negative slack time route. This proposed 500MHz clock frequency Linear Feedback Shift Register (LFSR) design operates for the required clock frequency without slack according to a Pre and Post clock tree synthesis (Pre-CTS & Post-CTS) performance report analyzed. Figure 1 depicts the critical steps that must be taken to attain clock tree synthesis before the implementation of Graphic Data System II (GDS II) for ASICs.

1.1 Contribution of this paper

- Based on the design aspect perspective, this study introduces and names the key principles of the linear feedback shift register.
- This paper discovers the consequence of clock tree synthesis.
- To adhere to timing limitations, setup and hold time analysis are also explored.

This research study is organized in the following order Section 2 is about the literature review Section 3 describes the proposed design specification of the Linear Feedback Shift Register(LFSR). Section 4 presents the physical design stream. Section 5 demonstrates the results and analysis. Finally, Section 6 concludes the paper with future directions

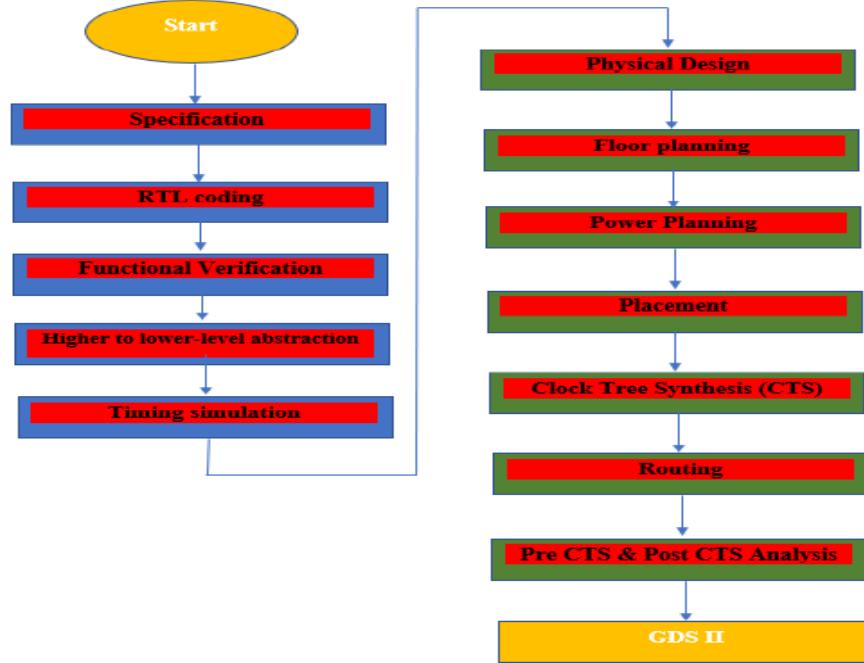


Fig.1. Stream for clock tree synthesis

2 Literature review

Application-specific integrated circuits benefit from timing-aware integration. Quality control for Reduce insertion delay, skew balance, duty cycle, pulse width, clock tree power consumption, signal integrity, and crosstalk via clock tree synthesis[3],[7]. Time impact affects optimization technique, placement, and global routing power consumption [8]. Latency and the transition in the timing path are two crucial factors that affect clock tree power; if latency is low, power consumption also less. Power usage recovers with smooth transitions[9].

The time frame required for the input to a flip-flop to be stable before a clock edge is known as setup time. In comparison to setup time, hold time covers events that take place after a clock edge. The hold time is the minimum period necessary for the input to a flip-flop to stay constant following a clock edge[10]. According to this study, after routing the design, there is zero worst negative slack during the assigned time per period iming analysis is influenced by clock lags, skew, cell

distribution throughout the path, slew, fanout, wire length, and net delay[12]. Optimization of physical design concerning result quality and clock net routing in a design process[13]. Innovative techniques can be used in System on Chip to ensure accurate clocking and synchronization at all levels of abstraction[14],[15].

3 Design specification of Linear Feedback Shift Register

In this research study, a shift register called a linear-feedback shift register (LFSR) has an input bit whose value is linearly correlated to its prior state. The most popular type of linear is used in this physical architecture for Cyber-Physical Systems. As an outcome, an LFSR is typically a shift register whose input bit is driven by the XNOR of some of the shift register's entire value. An LFSR fits into the state machine CPS class. The input bit of the shift register is a linear function of the preceding state.

Table 1. Requirement for the Design of LFSR

Sl.No	Specification	Values
1.	Clock frequency	500 MHz
2.	Amplitude	1 V
3.	Maximum Input Delay	1.0ns
4.	Maximum Output Delay	1.0ns
5.	Rise clock transition	0.1 V
6.	Fall clock transition	0.1 V

XOR and XNOR are solitary linear operations on single bits. The physical architecture of Linear feedback shift register(LFSR) with 500 MHz clock frequency, 1 Voltage amplitude with maximum input and output delay 1.0 ns, and clock transition of rise and fall edge 0.1v having 8-bit register designed. The output of the fifth and sixth register XNOR together to form feedback. Table 1 displays the specification of LFSR and Figure 2 displays the functional verification of LFSR.



Fig. 2. Functional verification of 8-bit Linear Feedback Shift Register

3.1 RTL Design and Verification

Transferring a higher level of abstraction to a lower level of abstraction that can be implemented during the synthesis stage which can be optimized for speed (timing), area, testability, and power. The operating parameters used in this design are Process 1ns, Voltage 0.9v, and Temperature 125°c. Three channel lengths are listed in Table 2 for examining lower-level abstraction. Figure 3 shows the LFSR's schematic diagram.

Table 2. Description of lower-level abstraction

Sl. No	Channel length	Technology	A figure of Merit (Power parameter)
1.	90nm	MOSFET	Leakage power-3.580nW Dynamic power-3122.096nW Total Power-3125.677nW
2.	45nm	MOSFET	Leakage power-789.399nW Dynamic power-3874.508nW Total Power-4663.907nW
3.	180nm	CMOS	Leakage power-15.46nW Dynamic power-17639.649nW Total Power-17655.113nW

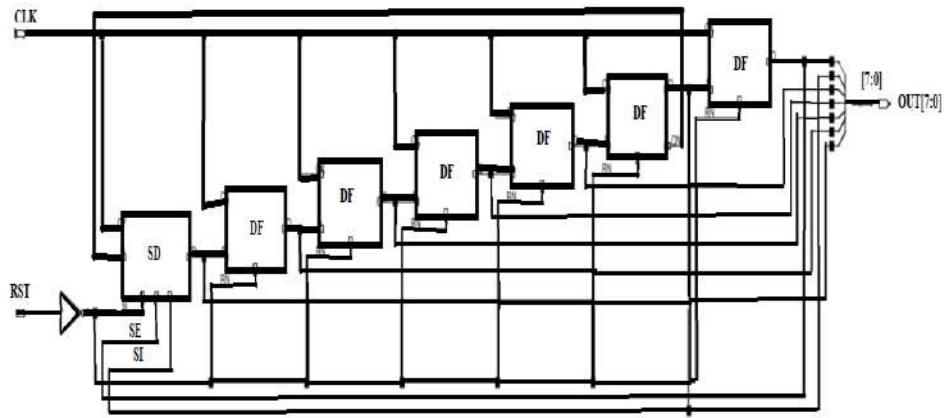


Fig. 3. Schematic diagram of LFSR

To select any one of the channel length constraints, 90 nm specification is used for proceeding with a physical architecture in this study. The following timing statement applies to the 90nm channel length MOSFET technology design and is shown in Table 3. The physical design of the LFSR was furthered using this report.

Table 3. Statement of Timing report

Sl. No	Pin	Fanout	Load(fF)	Slew(ps)	Delay(ps)	Arrival(ps)
1	Out_reg [5]/CK	NA	NA	0	NA	0
2	Out_reg [5]/QN	1	2.8	45	+390	390
3	Out_reg [0]/D	NA	NA	NA	+0	390
4	Out_reg [0]/CK	NA	NA	0	+260	650

4 Physical design stream

4.1 Floor planning

The physical design is essential because as technology advances, the intricacy of the designs and the size of the circuits both increase [16]. Floor planning is the first phase in the VLSI physical design pipeline. It determines the location and arrangement of each block on a chip to minimize chip area and connecting wire lengths. In this paper, the chip area is reduced by the core usage parameter. The floor planning specifications are provided in Table 4.

Table 4. Specification of the floor planning

Sl. No	Requirement	Values(μm)
1.	Aspect Ratio	0.8573
2.	Core Utilization	0.6999
3.	Core Margin I/O Boundary	8[core to left, right, top, bottom]

Figure 4(a) shows how the core occupies the whole floor planning area, with the design using roughly 70% of the core.

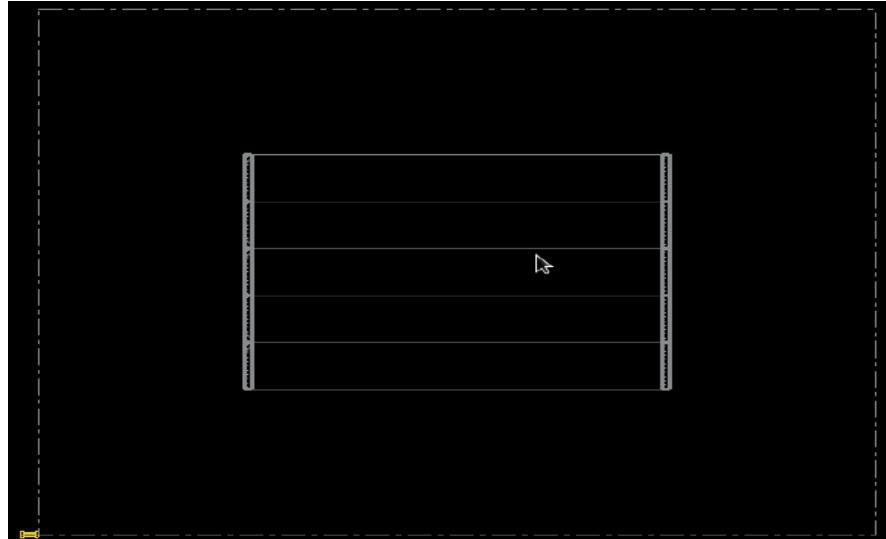


Fig.4. (a)

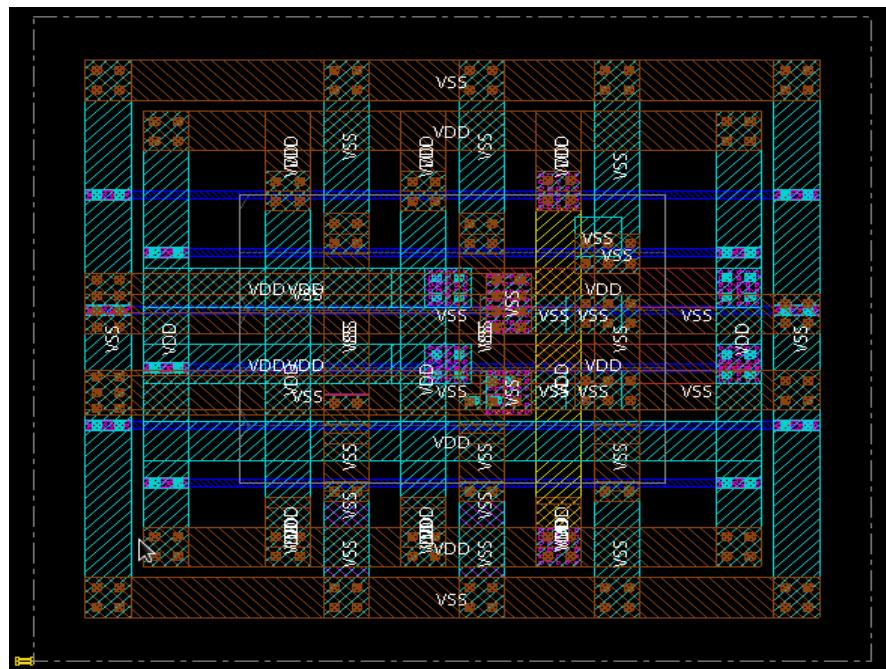


Fig.4. (b)



Fig.4. (c)

Fig.4.(a) Core Utilization of the floor plan(b) power planning of the design(c) placement design

4.2 Power planning

The core has been given power VDD and VSS pins for adding power rings and stripes with ring arrangement. With a width of 1.8 μm and a layer spacing of 0.5 μm , the top and bottom power rings in this design are made of Metal9(9), while the left and right power rings are made of Metal8(8). The design's power planning is depicted in Figure 4(b). Specifications for the power planning are listed in Table 5.

Table 5. Specification of the power planning

Sl. No	Ring	Layer	Width(μm)	Spacing(μm)
1.	Top	Metal9(9) (Horizontal)	1.8	0.5
2.	Bottom	Metal9(9) (Horizontal)	1.8	0.5
3.	Left	Metal 8(8) (Vertical)	1.8	0.5
4.	Right	Metal 8(8) (Vertical)	1.8	0.5

4.3 Placement design

The process of placing circuit devices on a die surface is described as placement. It is a crucial step in the design process for Very Large-Scale Integration (VLSI). Standard cells are placed on the core area shown in Figure 4(c), since it has an impact on a design's routability, performance, heat distribution, and to a lesser extent, power consumption. It is typically utilized following the logic synthesis phase and before the routing phase. The location has emerged as the key factor in circuit latency [17].

Because of these criteria Placement design was used for this research study. To improve circuit performance, placement information is crucial, especially in the early design phases. Placement techniques have recently been incorporated into the stages of logic synthesis and architecture design to perform physical synthesis and physical-aware architecture design [18], respectively. According to the Synopsys Design Constraint file (SDC file), which includes the required design attributes for each block's placement on the chip and its organization in the cadence tool.

4.4 Clock Tree Synthesis

To balance the clock delay to all clock inputs, the Clock Tree Synthesis (CTS) approach comprises the instinctive addition of buffers and inverters along the clock frequencies of the ASIC design. This is used to improve clock supply to all standard cells. CTS is used to diminish insertion latency and balance clock skew. A method for evenly dividing the clock throughout all sequential components of a VLSI design is called clock tree synthesis.

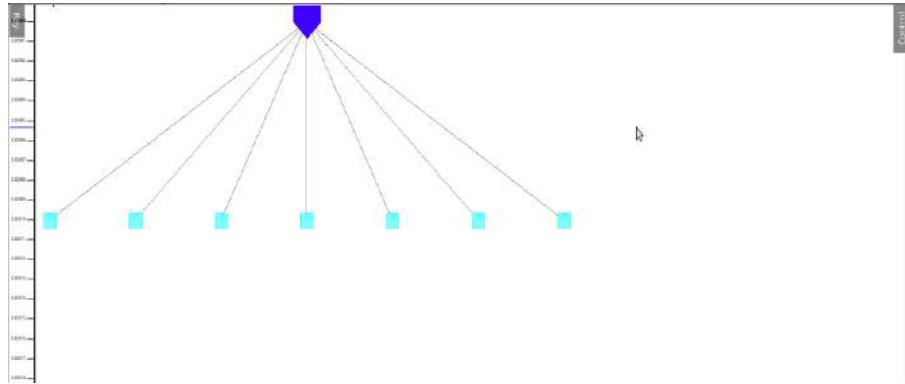


Fig. 5. Clock Tree Synthesis

Clock Tree Synthesis works to reduce skew and latency. Figure 5 illustrates the clock tree synthesis of LFSR physical architecture.

4.5 Routed power design

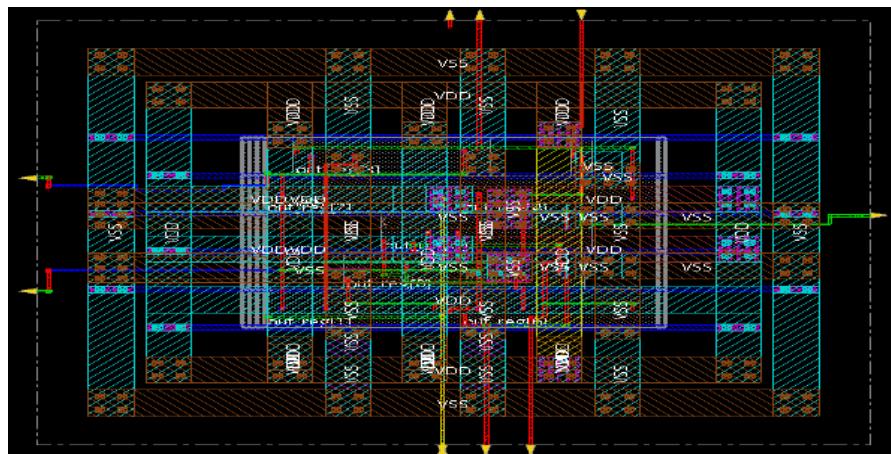


Fig.6. Final power-routed design

There is an effect on the routed power design that provides power to the LFSR standard cell. To enhance the timing path, it is essential to look at clock tree synthesis in this area [19],[20]. The final power routing architecture is shown in Figure 5.

5 Result and Analysis

The clock tree has a considerable impact on power consumption. Power consumption is influenced by the design's rise and fall times.

- Placement design or Timing design which has an impact on typical cell respects clock
- Special routing design has an impact on clock skew
- After physical routing optimized design was obtained. It is necessary to reevaluate the clock delay on this design

As a result of the Cadence Innovus tool, clock tree synthesis is crucial to these designs. The parameters for this synthesis are listed with setup time with worst-case and hold time with best-case scenarios in Table 6.

Table 6. Report of Pre and Post clock tree synthesis

Sl. No	Clock Tree Synthesis (CTS)	Worst Negative Slack (WNS) of Timing Path(ns) [register to register]					
		Placement Design		Routing Design		Optimized Design	
		Setup	Hold	Setup	Hold	Setup	Hold
		Worst	Best	Worst	Best	Worst	Best
1.	Pre CTS	1.339	0.011	1.332	0.010	1.333	0.010
2.	Post CTS	1.339	0.011	1.332	0.010	1.333	0.010

Table 6 implies that the zero slack on the worst-case timing path among these three designs provides Clock tree synthesis used to ensure that the operating clock frequency was applied to all of the design's standard cells.

6 Conclusion and Future directions

In this work, the clock tree synthesis for an 8-bit LFSR physical architecture is inspected. From the report, it is observed that the clock frequency is used in each LFSR cell without any slack. This architecture is suited for developing Field Programmable Gate Arrays in the

future from a machine learning perspective, as well as a Random Number Generator with Fibonacci and Galois number theory, Multibit LFSR, and Physically Unclonable Functions in Cyber-Physical Systems.

References

- [1]Chakravarthi, V.S. SoC Physical Design. In: A Practical Approach to VLSI System on Chip (SoC) Design. Springer, Cham (2022)
- [2]Circuit Modeling with Hardware Description Languages, Editor(s): Hubert Kaeslin, Top-Down Digital VLSI Design, Morgan Kaufmann, Pages 179-300,(2015)
- [3] Weiwei Shan, Wentao Dai, Liang Wan, Minyi Lu , Longxing Shi, Mingoo Seok , Jun Yang A Bi-Directional, Zero-Latency Adaptive Clocking Circuit in a 28-nm Wide AVFS System IEEE Journal of Solid-State Circuits, Vol. 55, No. 3, March (2020)
- [4] Clock Tree Synthesis in VLSI Physical Design (<http://www.ivlsi.com>)
- [5] A. B. Kahng, New directions for learning-based IC design tools and methodologies, 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 405-410,(2018)
- [6] Lee, E.A. The Past, Present and Future of Cyber-Physical Systems: A Focus on Models. Sensors 2015, 15, 4837-4869
- [7]A. B. Kahng, M. Kim, S. Kim and M. Woo, RosettaStone: Connecting the Past, Present, and Future of Physical Design Research, in IEEE Design & Test, vol. 39, no. 5, pp. 70-78, Oct. (2022)
- [8]C. -K. Cheng, A. B. Kahng, I. Kang and L. Wang, RePIAce: Advancing Solution Quality and Routability Validation in Global Placement, in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 38, no. 9, pp. 1717-1730, Sept. (2019)
- [9] H. Geng, T. Chen, Y. Ma, B. Zhu and B. Yu, PTPT: Physical Design Tool Parameter Tuning via Multi-Objective Bayesian Optimization, in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (2022)
- [10]Ahmed, M.V., Kariyappa, B.S. Optimization of Cloning in Clock Gating Cells for High-Performance Clock Networks. In: Mallick, P.K., Bhoi, A.K., Barsocchi, P., de Albuquerque, V.H.C. (eds) Cognitive Informatics and Soft Computing. Lecture Notes in Networks and Systems, vol 375(2022)
- [11] Nan Du, Heidemarie Schmidt, Ilia Polian,Low-power emerging memristive designs towards secure hardware systems for applications in internet of things,Nano Materials Science,Volume 3, Issue 2,Pages 186-204,ISSN 2589-9651,(2021)
- [12] S. Roy, Y. Ma, J. Miao, and B. Yu, A learning bridge from architectural synthesis to physical design for exploring power efficient high-performance adders, in Proc. ISLPED, pp. 1–6(2017)
- [13] K. Murugan, S. Baulkani,VLSI implementation of ultra power optimized adiabatic logic based full adder cell,Microprocessors and Microsystems,Volume 70 ,Pages 15-20,ISSN 0141-9331, (2019).
- [14] C. Zhuo, K. Unda, Y. Shi, and W.-K. Shih, From layout to system: Early stage power delivery and architecture coexploration, IEEE TCAD, vol. 38, no. 7, pp. 1291-1304,(2018)
- [15]] S. Hou, Y. Guo and S. Li, A Lightweight LFSR-Based Strong Physical Unclonable

- Function Design on FPGA, in IEEE Access, vol. 7, pp. 64778-64787, (2019)
- [16] Srinivasan, B., Venkatesan, R. Multi-objective optimization for energy and heat-aware VLSI floorplanning using enhanced firefly optimization. *Soft Comput* 25, 4159–4174 (2021)
- [17] Y. Lin, W. Li, J. Gu, H. Ren, B. Khailany and D. Z. Pan, ABCD Place: Accelerated Batch-Based Concurrent Detailed Placement on Multithreaded CPUs and GPUs, in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 39, no. 12, pp. 5083-5096, Dec. (2020)
- [18] Y. Ma, Z. Yu, and B. Yu, CAD tool design space exploration via Bayesian optimization, in Proc. MLCAD, pp. 1–6(2019)
- [19] X. Chen, G. Liu, N. Xiong, Y. Su and G. Chen, A Survey of Swarm Intelligence Techniques in VLSI Routing Problems, in IEEE Access, vol. 8, pp. 26266-26292,(2020)
- [20] T. Zhan, S. Z. Fatmi, S. Guraya and H. Kassiri, A Resource-Optimized VLSI Implementation of a Patient-Specific Seizure Detection Algorithm on a Custom-Made 2.2 cm² Wireless Device for Ambulatory Epilepsy Diagnostics, in IEEE Transactions on Biomedical Circuits and Systems, vol. 13, no. 6, pp. 1175-1185, Dec.(2019)

Detection of Arrhythmia via Electrical Activity of the Heart using AI techniques

Pramitha J¹, *Anitha Mary X¹,

¹Department of Robotics Engineering, Karunya Institute of Technology and Sciences, India

Corresponding author *anithamary@karunya.edu

Abstract. Electrocardiogram (ECG) consists of several waveforms such as the P wave, QRS complex, and T-wave. Various heart diseases can be diagnosed by observing the variations in the shape of each waveform and the distance between different peaks. There are 15 suggested classes for arrhythmia which are divided into 5 superclasses: Normal (N), Supraventricular ectopic beat (SVEB), Ventricular ectopic beat (VEB), Fusion beat(F), and Unknown beat(Q). The input signal is classified into these 5 categories using 1D Convolution Neural Network, Random Forrest Classifier, Decision Tree Classifier, and MPL Classifier. In the clinical diagnosis of cardiac illness, the ECG classification is crucial. The three main performance metrics – precision, recall, and F1 score are calculated for all 4 models and compared. The accuracy of the four models is also computed and compared. The One Dimensional Convolution Neural Network achieved 98%accuracy, 90% macro average, and 98% weighted average. The Decision Tree Classifier achieved 95% accuracy, 80% macro average, and 95% weighted average. The Random Forest Classifier achieved 97% accuracy, 86% macro average, and 97% weighted average. The MLP Classifier achieved 98%accuracy, 87% macro average, and 97% weighted average.

Keywords: Electrocardiogram, classes, machine learning algorithm, performance metrics

1 Introduction

Cardiovascular disease is caused due to various conditions. Some of these conditions might be a heart attack, stroke, heart failure, arrhythmia, and other such conditions. Arrhythmia is caused when the heart beats too slowly, too quickly, or irregularly. Detection of Arrhythmia at an early stage helps greatly with the treatment [9,13].

An electrocardiogram (ECG) records the electrical activity of the heart. The electrodes are placed on the skin to record the electrical changes brought on by the heart muscle's depolarization and repolarization during each cardiac cycle. These signals are recorded by a machine and are looked at by a doctor to see if they're unusual.

The deflections in each cardiac cycle are represented in the form of the P wave, QRS complex, and the T-wave. The P wave represents the arterial depolarization, the QRS complex represents the ventricular depolarization, and the T-wave represents the ventricular repolarization.

- Pwave –atrial depolarization–atria contracts
- QRS complex–ventricles depolarization–ventricles contracts
- T-wave–ventricular repolarization–ventricles relax

Previously, many methods have been put forth to find arrhythmia. Majority of these methods relied on classification algorithms like Support Vector Machines (SVMs) [1] and digital signal processing (DSP) techniques. Deep Neural Network-based Machine Learning algorithms have proliferated across all fields and can also be used in detecting arrhythmia.

2 Related Works

The different kinds of techniques and classification algorithms for the classifica-

tion of arrhythmia are reviewed from selected articles as shown in Table 1.

Table 1: literature review

Sl. No.	Author	Inference
1	Celinet.al., (2018) [1]	Results show that the accuracy of the SVM, Adaboostand ANN classifies 87.5%, 93% and 94%
2	Luz et.al., (2016) [8]	MIT-BIH database's heartbeats were suggested to be split into two sets so that the database would be more consistent with reality.
3	Romdhaneet.al., (2020) [12]	Classification is done through 1DConvolutionalNeuralNetwork
4	Martinez-Useros,C.,et al.,(2022) [3]	Higher accuracy was given for aortic disease detection
5	Wu,Mengze, et.al., (2021) [4]	The R-R peak is detected and is used for classification

3 Methodology

3.1 Objective:

To detect cardiovascular diseases, the identification and classification of ECG sig-

nals are necessary. Manual identification of ECG heart-beat classes by cardiologists is time-consuming. Therefore, various artificial intelligence techniques are being used for identifying ECG characteristics. The primary goal is to categorize the ECG into its five main classes. The following are the main design objectives:

1. Classify the ECG into its 5 classes (N, SVEB, VEB, F, Q)
2. Use 1D Convolution Neural Network, Random Forrest Classifier, Decision Tree Classifier, and MPL Classifier to classify the ECG

Figure 1 shows the generalized diagram of proposed work for ECG classification.

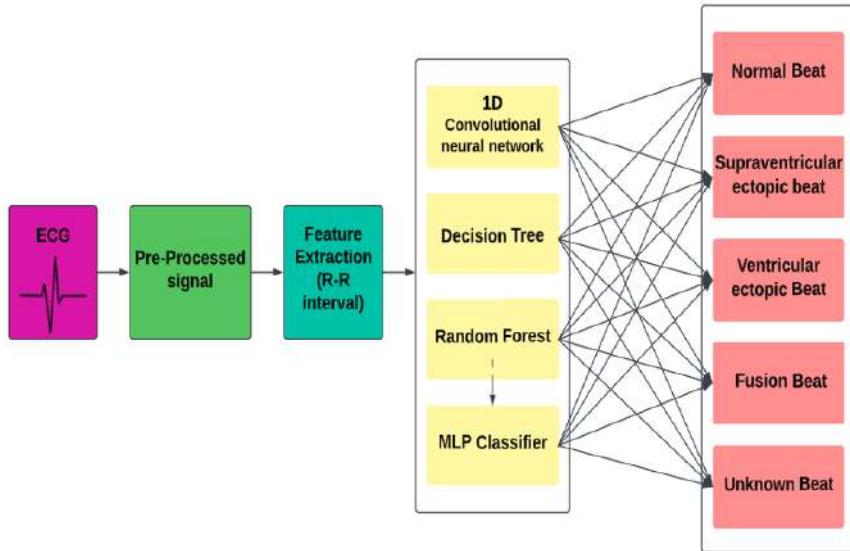


Fig 1: ECG classification using various machine learning algorithms

3.2 Dataset

The dataset for MIT-BIH arrhythmias is used in this study. The dataset compiles 48 patients' 30-minute raw ECG recordings, 25 of who are female and 22 of whom are male. Each record is made up of two lead signals: the upper signal, which is a modified limb lead II produced by placing lead on the chest, and the bottom signal, which is normally a modified lead V1[8]. The dataset's records are each sampled once per

360 Hz. Significant QRS complexes can be seen in the upper signal MLII, and the database is primarily split into 15 subclasses and five superclasses of heart arrhythmias [2]. The details are given below.

- 109446 samples are available in the database[11]
- These samples are classified into 5 categories
- The sampling frequency is 125Hz
- The data source is Physionet's MIT-BIH Arrhythmia Dataset
- Super Classes:[Normal beat 'N':0,Supraventricular ectopic beat 'S':1, Ventricular ectopic beat 'V':2,Fusion beat 'F':3, Unknown beat 'Q':4]

3.3 Pre-processing and Feature extraction

The dataset's raw ECG signals are added together with interferences such as EMG, power frequency, and noise. The data must be free of these interferences to accurately detect heartbeats with arrhythmias. To achieve this, the raw ECG signals have been de-noised using the Discrete-Wavelet-Transform. The denoising filter lessens the QRS complex's signal distortion, allowing for a clearer expression of the complex's signal in the detection of RR intervals[10]. Additionally, the preprocessed dataset was split into 20% of the data for testing and 80% for training the model [6] to accurately examine the proposed approach. The collection consists of 48 records, each of which represents a distinct patient.

4. Arrhythmia Classification

The main goal of this study is to present four algorithms to classify arrhythmia with the available MIT-BIH database using artificial intelligence techniques.

4.1 Convolutional neural network(CNN)

Convolutional neural network (CNN) is one of the first effective deep learning algorithms, mostly used for the classification of images, video, texts, and speech. The three layers present in CNN are convolution, pooling layers and activation layers [13]. The main advantage of CNN is that it automatically observes and extracts the most important feature from the given set of inputs.

4.2 MLP Classifier

A multilayer perceptron (MLP) is an artificial neural network that generates outputs from a set of inputs. A neural network is used by the multi-layer perceptron classifier to carry out the classification operation. MLP makes use of back propagation for training the network.

4.3 RandomForestAlgorithm

Random Forest is a machine learning algorithm that falls under the supervised learning technique. Random Forest is a classifier that uses numerous decision trees on different subsets of the provided dataset and considers the average to increase the dataset's predictive accuracy. The random forest decides based on the results from each decision tree instead of relying on just one, and the ultimate result is determined by the majority of predictions. The greater number of trees in the forest, the higher the accuracy. When each tree in the forest casts a unit vote for the most predicted class, classification takes place. The categorization with the greatest votes among all the trees is then selected by the Random Forest classifier. The Random Forest technique is suitable for long-term ECG beat categorization since it is fast, performs well, and requires no cross-validation. It exhibits good performance on a variety of real-world issues since it is not overfitted and is not sensitive to data set noise. An overfit model observes all the features of the training dataset and performs well but produces very poor results for the testing dataset.

4.4 Decision tree

A decision tree is a tree-structured classifier. The features of a dataset are described by the internal nodes, the decision rules are described by the branches and the results are represented by each leaf node. Decision Node and Leaf Node are the two types of nodes present in this classifier. The decisions are made by the decision node and they have multiple branches [7], whereas the leaf nodes hold those results and have no further branches. The results of the decisions are generated based on the features of the dataset given. A question is asked and based on the an-

swer (Yes/No), the decision tree branches and forms a tree-like structure.

5. Results and Discussions

It is inferred that the accuracy of One Dimensional Convolutional Neural Network is 98%, the Decision Tree Classifier is 95%, Random Forest Classifier is 97% and MLP Classifier is 98%. Further, the precision, recall, and F1 score for every model have also been tabulated and plotted.

The 1DCNN achieved 98% accuracy, 90% macro average, and 98% weighted average. The Decision Tree Classifier achieved 95% accuracy, 80% macro average, and 95% weighted average. The Random Forest Classifier achieved 97% accuracy, 86% macro average, and 97% weighted average. The MLP Classifier achieved 98% accuracy, 87% macro average and 97% weighted average.

4.1 Interpretation of performance:

		Prediction made	
		True Positive (TP)	False Positive (FP)
Actual result	True Negative (TN)		
	False Negative (FN)		

True Positive (TP): These accurately predicted positive values show that both the actual result and the prediction made are true.

True Negative (TN): These accurately predicted negative values show that both the actual result and the prediction made are true.

False Positive (FP): This occurs when the actual result is false but the prediction made suggests that it is true

False Negative (FN): This occurs when the actual result is true but the prediction made suggests that it is true

Accuracy:

Accuracy is the sum of all the accurately predicted values to the total number of observations [5].

$$\text{Accuracy} = \frac{\text{Accurately predicted values}}{\text{Total number of observations}}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Precision:

Precision is given by accurately predicting positive values to the total predicted positive values.

$$\text{Precision} = \frac{\text{Accurately predicted positive values}}{\text{Total predicted positive values}}$$

$$\text{Precision} = \frac{TP+TN}{TP+FP}$$

Recall:

The recall is given by the total number of accurately predicted positive values to the total number of values in the actual results.

$$\text{Recall} = \frac{\text{Accurately predicted positive values}}{\text{Actual results}}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

F1 Score:

To calculate the F1 Score, the average precision and recall are considered.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Table 2 shows the performance metrics for 1D CNN. It is observed that precision, recall, and F1 score values are high for [Q].

Table 2 Performance Report of 1D CNN

Classes of Arrhythmia	PRECISIO N	RECAL L	F1- SCORE
[N] 0	0.98	1	0.99
[S] 1	0.91	0.73	0.81
[V] 2	0.99	0.88	0.93
[F] 3	0.75	0.81	0.78
[Q] 4	0.99	0.98	0.99

Table 3 shows the performance metrics for the Decision Tree. It is observed that precision, recall, and F1 score values are high for [N].

Table 3 Performance Report of Decision Tree

Classes of Arrhythmia	PRECISION	RECALL	F1-SCORE
[N] 0	0.98	1	0.99
[S] 1	0.62	0.63	0.63
[V] 2	0.86	0.86	0.86
[F] 3	0.58	0.62	0.6
[Q] 4	0.94	0.94	0.94

Table 4 shows the performance metrics for Random Forest. It is observed that preci-

sion, recall, and F1 score values are high for [N] and [Q].

Table 4 Performance Report of Random Forest

Classes of Arrhythmia	PRECISION N	RECALL LL	F1- SCORE
[N] 0	0.97	1	0.98
[S] 1	0.99	0.58	0.73
[V] 2	0.98	0.86	0.92
[F] 3	0.88	0.59	0.7
[Q] 4	1.00	0.94	0.96

Table 4 shows the performance metrics for MLP. It is observed that precision, recall, and F1 score values are high for [N] and [Q].

Table 5 Performance Report of MLP classifier

Classes of Arrhythmia	PRECISION	RECALL	F1-SCORE
[N] 0	0.98	1	0.99
[S] 1	0.89	0.64	0.75
[V] 2	0.96	0.9	0.93
[F] 3	0.87	0.6	0.71
[Q] 4	0.99	0.96	0.98

Figure 2 shows the graphical representation of performance metrics for 1D CNN, Decision Tree (Figure 3), Random Forest (Figure 4), MLP Classifier (Figure 5)

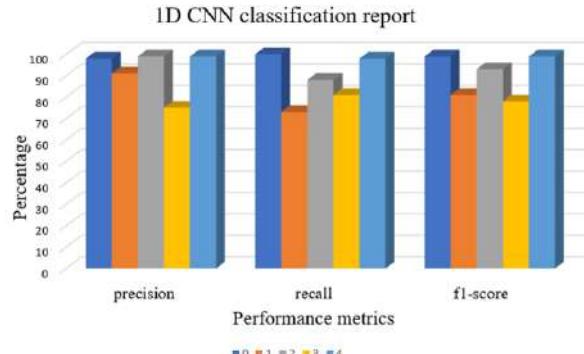


Fig 2: Graphical view of performance report of 1D CNN

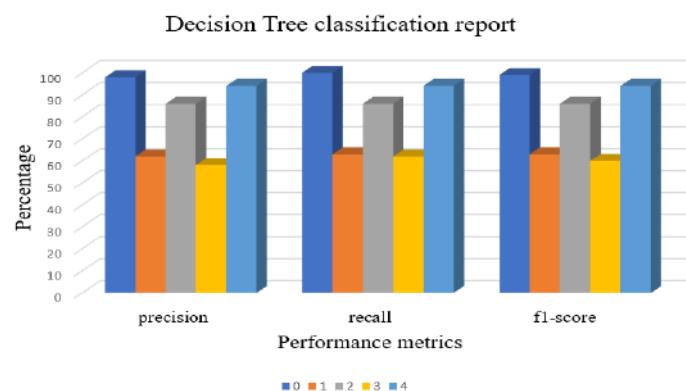


Fig 3: Graphical view of performance report of Decision Tree

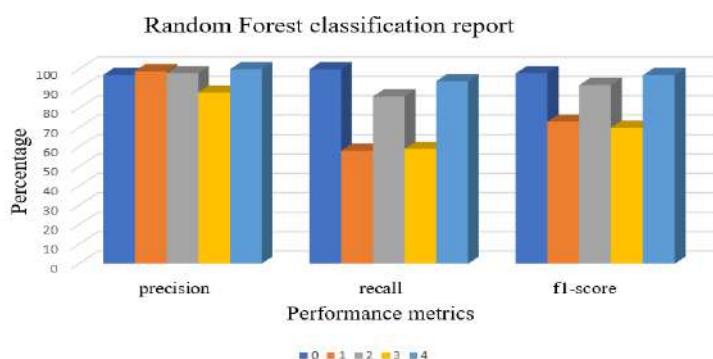


Fig4: Graphical view of performance report of Random Forest

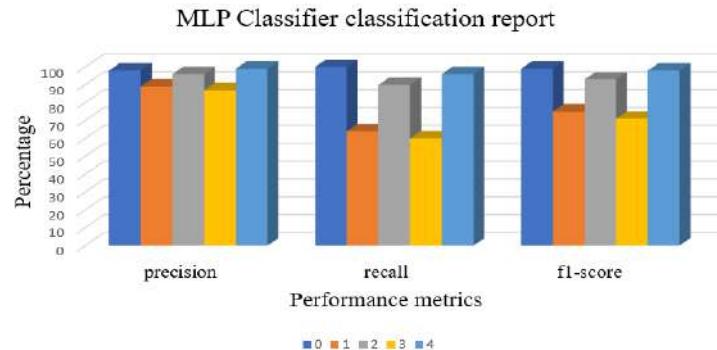


Fig 5: Graphical view of performance report of MLP Classifier

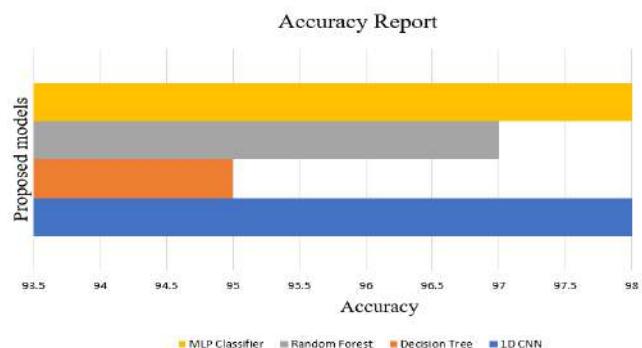


Fig 6: Graphical view of accuracy report of various ML algorithms

Table 6 and Fig 6 show that the accuracy of the 1D -CNN is 98% as they automatically learn the key features of the given training dataset. It is based on three convolution, max pooling, and dense layers which automatically extract distinguishable nonlinear features from the ECG signals and automatically classify them into five different classes. The accuracy of Decision Tree and Random Forest Classifiers are 95% and 97% respectively. The MLP Classifier has achieved an accuracy of 98% because of its hidden layers and uses the back propagation technique for its training.

Table 6. Comparative study on the accuracy of various ML models

MODEL	ACCURACY
1D CNN	98
decision tree	95
RandomForest	97
MLPClassifier	98

5. Conclusion

The precise analysis of heartbeats via ECG is crucial for treating cardiovascular disorders since they constitute a serious threat to human life. Medical professionals must invest a lot of time and money in manually analyzing ECG readings. To do this, the emphasis has switched from manual analysis to automatic identification of anomalies in heartbeat.

Four models have been used for classifying ECG signals – One Dimensional Convolution Neural Network, Decision Tree Classification, Random Forest Classifier, and MLP Classifier. One major obstacle to developing a completely automatic classification of heartbeats using ECG is that the database availability is limited. Arrhythmias (especially ventricular arrhythmias) were majorly detected in patients diagnosed with aortic stenosis, pulmonary stenosis, or ventricular septal defect. "Serious arrhythmias" were most frequently seen in patients with aortic stenosis who also had a higher chance of sudden death. Severe ventricular arrhythmias in aortic stenosis may be associated with left ventricular dysfunction and should be taken into account when deciding if surgery should be performed. Thus, this study can be extended to train the models to extract more features to detect ventricular arrhythmias that can also be used to detect Aortic stenosis. The suggested method

can be improved and tested in the future to cope with real-time data. Additionally, the model can be made simpler to be used with embedded systems.

Reference

1. Celin, S., Vasanth, K.: ECG Signal Classification Using Various Machine Learning Techniques. *J Med Syst* **42**, 241 (2018).
2. Kanani, Pratik, and MamtaPadole: Electrocardiogram heartbeat classification based on a deep convolutional neural network and focal loss, *Computers in Biology and Medicine*, Volume 123, 103866, ISSN 0010-4825, (2020)
3. Martínez-Useros, C., et al.: Ventricular Arrhythmias in Aortic Valve Disease: A Further Marker of Impaired Left Ventricular Function, *International Journal of Cardiology*, vol. 34, no. 1, pp. 49–56, 10.1016/0167-5273(92)90081-d. (2022)
4. Wu, Mengze, et al.: A Study on Arrhythmia via ECG Signal Classification Using the Convolutional Neural Network, *Frontiers in Computational Neuroscience*, vol. 14, 10.3389/fncom.2020.564015. (2021).
5. Das, Manab Kumar, and Samit Ari. : ECG Beats Classification Using Mixture of Features. *International Scholarly Research Notices*, vol. 2014, pp. 1–12, 10.1155/2014/178436, (2014)
6. Kanani, Pratik, and MamtaPadole.: ECG Heartbeat Arrhythmia Classification Using Time-Series Augmented Signals and Deep Learning Approach. *Procedia Computer Science*, vol. 171, 2020, pp. 524–531, 10.1016/j.procs.2020.04.056. (2020)
7. Lu, Peng, et al.:Identification of Arrhythmia by Using a Decision Tree and Gated Network Fusion Model. *Computational and Mathematical Methods in Medicine*, vol. 2021, pp. 1–13, 10.1155/2021/6665357, (2021)
8. Luz, Eduardo José da S., et al.: ECG-Based Heartbeat Classification for Arrhythmia Detection: A Survey. *Computer Methods and Programs in Biomedicine*, vol. 127, pp. 144–164, 10.1016/j.cmpb.2015.12.008, (2016)

9. Mohebbanaaz, et al.: Classification of ECG Beats Using Optimized Decision Tree and Adaptive Boosted Optimized Decision Tree. *Signal, Image and Video Processing*, 10.1007/s11760-021-02009-x. (2021)
10. Pandey, Saroj Kumar, et al.: Patient-Specific Machine Learning Models for ECG Signal Classification. *Procedia Computer Science*, vol. 167, 2020, pp. 2181–2190, 10.1016/j.procs.2020.03.269.,(2020)
11. Philip de Chazal, M. O'Dwyer and R. B. Reilly: Automatic classification of heartbeats using ECG morphology and heartbeat interval features, in *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 7, pp. 1196-1206, (2004)
12. Romdhane, TaissirFekih, et al. :Electrocardiogram Heartbeat Classification Based on a Deep Convolutional Neural Network and Focal Loss. *Computers in Biology and Medicine*, vol. 123, p. 103866, 10.1016/j.combiomed.2020.103866. (2020)
13. Zhang, Dengqing, et al.: An ECG Heartbeat Classification Method Based on Deep Convolutional Neural Network. *Journal of Healthcare Engineering*, vol. 2021, pp. 1–9, 10.1155/2021/7167891, (2021)

Utilizing Deep Convolutional Neural Networks for Image-Based Plant Disease Detection

Saha Reno¹[0000–0003–1897–9002], Marzia Khan Turna¹[0000–0001–8080–4292],
Sheikh Tasfia²[0000–0002–0436–1087], Md. Abir³[0000–0003–4623–1624], and Anusha
Aziz⁴[0000–0002–7829–3769]

¹ Bangladesh Army International University of Science and Technology, Cumilla,
Bangladesh

reno.saha39@gmail.com

marziakhanturna@gmail.com

² Military Institute of Science and Technology, Dhaka, Bangladesh
sheikhtasfia0602@gmail.com

³ University of Erlangen Nuremberg, Erlangen and Nuremberg, Bavaria, Germany
abirkhan88821@gmail.com

⁴ Bangladesh University of Business and Technology, Dhaka, Bangladesh
anusha.cse.mist@gmail.com

Abstract. Automatic leaf disease detection is a challenging problem in smart agriculture due to the variations in appearance and the complicated backgrounds of plant diseases. Designing a deep convolutional neural network model that extracts visual illness features from the photos and then recognizes the diseases based on the retrieved features is a standard solution to this problem. This method works well when the background is simple, but it has poor accuracy and robustness when the background is complex. The primary factor in identifying leaf-based plant diseases is the color information of sick leaves. Most existing approaches for diagnosing plant diseases depend on the expert diagnosis, which inevitably results in field management and crop disease control behind the times. In this paper, our methodology is based on Deep CNN which can be applied to solve these problems to improve the speed and accuracy of disease classification and recognition of plant diseases. We mainly focus on a Deep CNN-based model with fine-tuning vegetable leaf diseases dataset. Transfer learning models implemented with famous pre-trained models such as VGG-16 and ResNet-50 are compared with our proposed model. Our proposed model outperforms various existing solutions with an mAP (Mean Average Precision) accuracy of 99.80%.

Keywords: Plant disease detection · Deep convolutional neural network · Object detection · Machine learning · ResNet-50 · VGG-16 · YOLOv5.

1 Introduction

Plant disease symptoms often appear on leaves, buds, fruits and young twigs. Fruit is destroyed or wasted as a result. Additionally, these diseases cause the

development of fresh infections and the spread of the illness due to factors including seasonal weather. Due to this, it is crucial to identify the disease beforehand and take the appropriate measures to prevent it from spreading to other trees. As a result, the primary concern in agriculture is the fight against plant pests and diseases. Different plant diseases exist, and they can all cause harm to the economy, society, and environment. In this situation, a fast and correct diagnosis of plant diseases is crucial to preventing the loss of agricultural output and quantity. Plant disease detection is typically done by hand. These procedures are carried out by specialists such as agricultural engineers and researchers, first by visualization and then in the research lab. These conventional techniques are complex and time-consuming procedures as these techniques are computationally costly, the training takes a large amount of training data, and each training example demands extensive calculations and space.

Deep convolutional neural networks (CNNs) have recently made significant advancements in these fields with deep learning and visual object recognition. The ability to extract features without using segmented approaches is one of the leading facilities of architecture and methods developed in those areas. As a result, using these methodologies and models in practical applications is simple. Numerous research on the detection of plant diseases has found excellent efficiency when CNNs are utilized as fundamental deep learning methods. CNN employs convolution and pooling techniques to subsample the input prior to implementing an activation function, in which all of the levels are partly linked and buried, with the last layer being the output layer.

In summary, research on plant disease can be said to have used conventional image processing techniques for recognition and have obtained specific outcomes and good disease accuracy acknowledgment. However, there are still flaws and restrictions, such as follows:

1. The relationships and procedures used in research are complex, highly individualized, labor-intensive, and time-consuming.
2. It is extremely reliant on the spot segment.
3. Largely dependent on the extraction of the artificial feature. Extraction of artificial features can be done in four methods: (i) SuperPoint: Self-Supervised Interest Point Detection and Description, (ii) D2-Net: A Trainable CNN for Joint Description and Detection of Local Features, (iii) Image Feature Matching Based on Deep Learning, and (iv) Deep Graphical Feature Learning for the Feature Matching Problem.
4. The disease recognition performance is quite challenging to test.

Plant diseases greatly influence the productivity and quality of plants and pests [13]. Deep learning technology has emerged in recent years to study the recognition of plant diseases and has significantly outperformed conventional approaches in the field of digital image processing in recent years [16]. Researchers' top research concerns now center on identifying plant diseases and pests using deep learning technologies [12]. We need to put a lot of effort into classifier design, expressing innovative end-to-end qualities in the picture [3]. To begin, we

must provide the neural network's input layer with arrays representing the image's pixels. The extraction of features from the image is done by a number of hidden layers, including the convolution layer, the ReLU layer, and the pooling layer. The object in the image is finally identified by a fully connected layer. Due to these qualities, deep learning technology is the extensive emphasis given to detecting plant diseases, and it is becoming a popular subject for investigation [7].

Deep CNN techniques are currently widely used in many computer vision applications, and predicting plant diseases is typically considered a specialized agricultural application [6]. The number of agricultural crop pests and disease samples needs to be improved. Self-collected data sources are less than open standard sources and require laborious labeling [4]. The problem of tiny samples is the most pressing in detecting plant diseases in contrast to the more than 14 million sample data in ImageNet datasets [15]. Only a few or a dozen training data are often obtained due to the low incidence and expensive cost of various plant diseases, which restricts the use of Deep CNN techniques in detecting plant diseases [10]. There are now three possible solutions to the small sample problem.

In this study, a Deep CNN-based system for plant disease classification is proposed and implemented along with the evaluation.

1. A Deep CNN-based system for plant disease classification is proposed and implemented along with the evaluation.
2. We used a dataset consisting of actual plant disease images collected from various online sources like Kaggle and Google Scholar.
3. First, we extract deep features from this dataset using deep learning architectures: YOLOv5, VGG-16, and ResNet-50.
4. After that, we used epoch=100 to evaluate the implementation.
5. Finally, we compare each of the algorithms. We get the best result in the YOLOv5 algorithm compared to VGG-16 and ResNet-50.

2 Literature Review

Numerous research articles have been written about the use of deep learning in agriculture, which is crucial to the well-being of plants and crops, deep learning literature review:

After the algorithm completed the test to ensure, Joshi, R.C. proposed VirLeafNet: Automatic analysis and viral disease diagnosis using deep learning in the Vigna mungo plant. All the models achieved high validation accuracy and produced testing accuracy for VirLeafNet-1, VirLeafNet-2, and VirLeafNet-3 as 91.234%, 96.429%, and 97.403% on different leaf images, respectively. However, several bacterial, fungal, and viral illnesses were not considered [5].

Saumya Yadav; Analysis of peach leaf bacteriosis with Deep CNN for disease prediction. This project contrasts the outcomes of the CNN technique with the imaging methodology. This study's model architectures created using various deep learning techniques performed best, correctly detecting the associated

peach leaf [bacteria-free and healthy] in 0.185 seconds for each image. An accuracy of 98.75% is achieved but our system increases the accuracy to 99.80% using YOLOv5 [19].

Vaibhav Tiwari created a paper on multiclass plant disease recognition and classification using dense CNN based on a leaf image dataset. The proposed model of DADCNN-5 outperforms the current ML architecture and conventional CNN designs, according to experimental results presented in this research, and obtained 97.33% accuracy. The results were 96.57%, 99.94%, and 0.063% for sensitivity, specificity, and false positive rates, respectively. The module completes the training procedure in around 3235 minutes and achieves an accuracy rate of 99.86%. Corn Blight, Corn Common Rust, Corn Gray Spot and Corn Healthy were not included and considered in their research. [18].

M. Lv, Maize utilizes DMS-Robust Alexnet and Leaf Disease Identification Based on Feature Enhancement. In this study, Alexnet's recognition accuracy reaches 91.83%, demonstrating a trend for accuracy to converge. DMS-Robust Alexnet has a recognition accuracy of 98.62%. Only diseases related to corn leaf are considered and classified using Alexnet and DMS-Alexnet falls under the original CNN, hence cannot perform better than Deep CNN [8].

Ishrat Zahan Mukti's research focuses on using ResNet-50 to detect plant diseases by transfer learning. The most remarkable performance of their suggested model was 99.80% training accuracy and 100% validation accuracy. The proposed model is based on ResNet50 which cannot outperform YOLOv5 with respect to training loss and training time per epoch. Also, YOLOv5 requires much less computational power compared to ResNet50 [9].

Xiao Zhang Detecting diseases on maize leaves with enhanced deep convolutional neural networks. In this research, improved Cifar-10 and GoogleNet models are provided and their performance is compared to AlexNet and VGG to achieve better/more transparent detection of plant diseases. Updated versions of these models delivered a remarkable accuracy of 98.9%. This research focuses only on detecting maize leaf disease while our system considers a wide range of plant diseases and achieves almost similar accuracy [20].

Su, J. suggested multispectral UAV aerial images to monitor Wheat yellow rust. The accuracy of the Random Forest (RF) classifier combined with the multispectral imaging approach employed in this study to detect wheat illness was 89.3%. Random Forest is considered less efficient and performs poorly compared to Deep CNN architectures. [14].

This research used pre-trained weights from the popular YOLOv5 model to build a deep convolutional neural network utilizing the Transfer Learning methodology. These additional layers in our suggested model primarily aid in feature extraction with minimal computational expense. Additionally, adjustments have been made to improve the detection's accuracy. The dataset for our research shows an image of 11 different types of diseases combined with images of plant leaves. Our method for recognizing plant diseases is described in the subsection that follows.

3 Methodology

Numerous features have been proposed to increase Deep Convolutional Neural Network (CNN) accuracy. It is necessary to evaluate these feature combinations in practice on massive datasets and to support the outcome theoretically. In CNN and Deep CNN, every model architecture, dataset, and tasks have some aspects, such as batch, residual connections and normalization, which are relevant to fixed models with specific issues. Data normalization is achieved by 1NFm, which makes sure that no items within a group are repeated, every record is distinct and each entry must contain only one value for each cell.



Fig. 1. Samples of Image-Based Dataset for Vegetable Leaf Diseases

3.1 Dataset

There are three categories of plants represented in our research dataset. We have collected plant disease datasets from the Roboflow universe, Kaggle, and articles from Google Scholar [1][11][2]. An image dataset that contains images of both healthy and diseased plants. Different classes of leaves were used to train the proposed model. A CNN model was trained to learn more about the traits and to be able to determine between one class and another and their respective health status. 5261 training images and 1510 validation images of diseased and

healthy plants image were used for this research. Sample statistics of our dataset can be found in Fig 1.

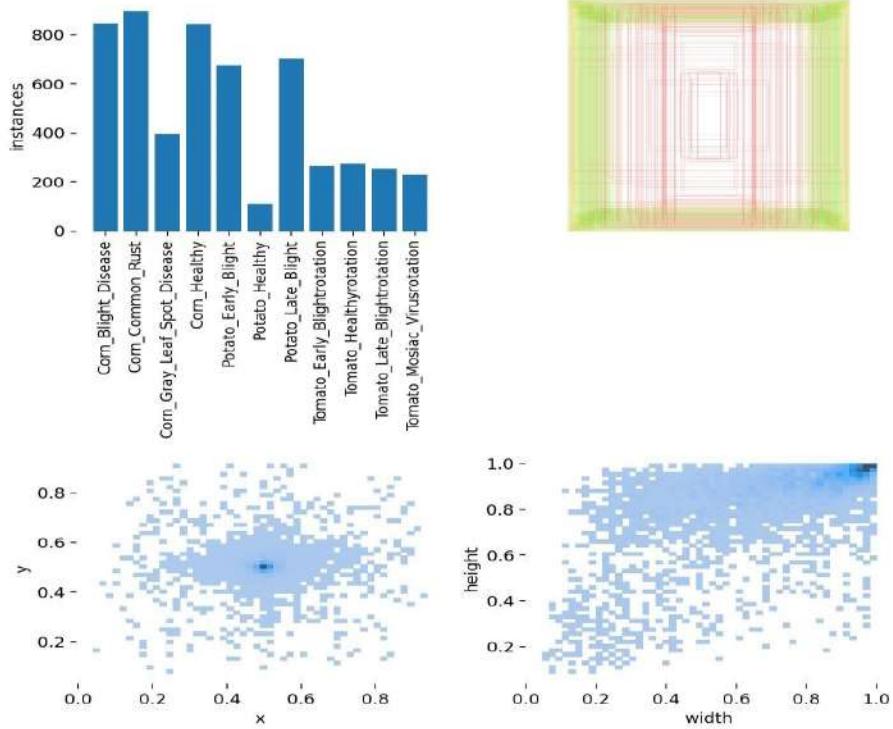


Fig. 2. Class Instances Based on Vegetable Leaf Diseases Dataset Labeling

3.2 Image Preprocessing and Labelling

In image preprocessing, the image data needed for images to be classified is generated. A method for automating and processing many transactions as a homogeneous group is batch processing. In this process, large volumes of data must be handled all at once. Batch processing techniques employ geometric adjustments to images, such as image rotation, translation, and scaling. We utilize Roboflow for labeling and picture preprocessing on our dataset, which is illustrated in Fig. 2.

We have scaled all photos down to a resolution of 416*416 pixels during the preprocessing processes. It must guarantee that each image has the exact resolution. We must label or classify images using a keyword search to search them quickly. All transcribed images were taken out of the dataset during this time.

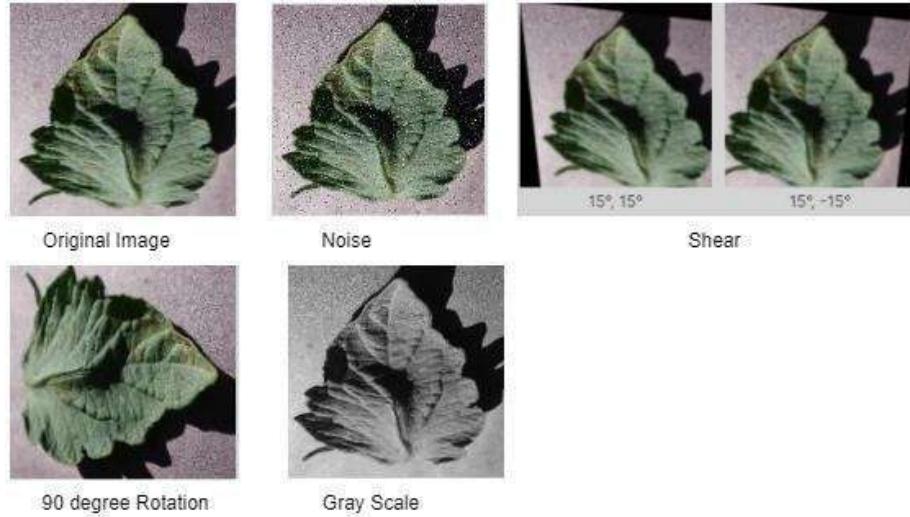


Fig. 3. Sample Augmentation of the Image-Based Dataset

3.3 Augmentation Process

Our vegetable leaf diseases prediction dataset is extended using different methods to add more images. It aids in preventing the issue of overfitting during the training procedure. When we train our dataset, our proposed model learns the data from the overall pattern of the dataset, and overfitting is the issue. Applying specific modifications to the images such as rotation, width shift, shear range, height shifting and horizontal flipping, noise addition, and gray scaling from original image augmentations were carried out. Fig. 3 illustrates a sample augmentation of the dataset utilized in our research.

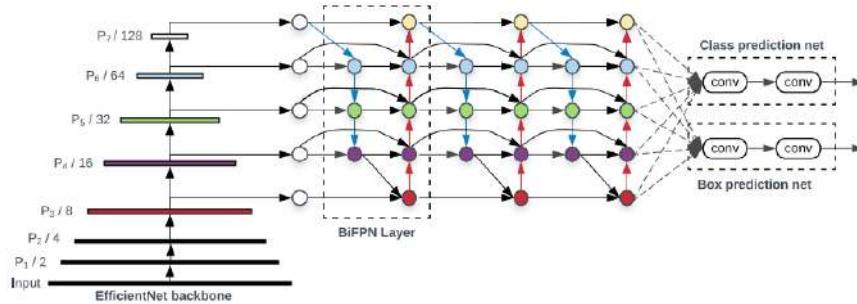


Fig. 4. Architecture of YOLOv5 EfficientNet

3.4 Disease Classification Using YOLOv5

The structure of the developed YOLOv5 model, as shown in Fig. 4, was divided into two parts for this article. A training model is one, while a detection model is another. 5261 images from the dataset of vegetable leaf diseases were applied to the training model. The detection method analyzed input images from the Vegetable Leaf Disease dataset and processed a prediction score of eleven classes: Corn Blight Disease, Corn Gray Leaf Spot Disease, Corn Common Rust, Potato Early Blight, Corn Healthy, Potato Healthy, Potato Late Blight, Tomato Early Blight Rotation, Tomato Healthy Rotation, Tomato Late Blight Rotation, and Tomato Mosaic Virus Rotation. The output images were then provided with their predicted classes and detection value.

YOLO is a quick object detection model. In relation to its size, it performs well and keeps improving. The most recent version of YOLOv4 is YOLOv5. YOLOv3 is an expansion of the YOLOv5 repository. According to [17], YOLOv5 is faster than its earlier iterations, YOLOv4 and YOLOv3. YOLOv5 is simple to use, configure, implement, and test, yet substantial research has been done on it. YOLO is a real-time object identification model that recognizes video and image objects. The YOLOv5 recognizes and categorizes the images and may find several images inside of one image. The YOLO debuted in 2015 and is extremely fast in real-time object identification.

A use case for which YOLOv5 is intended involves the creation of features from input images for disease prediction. First off, an image is divided

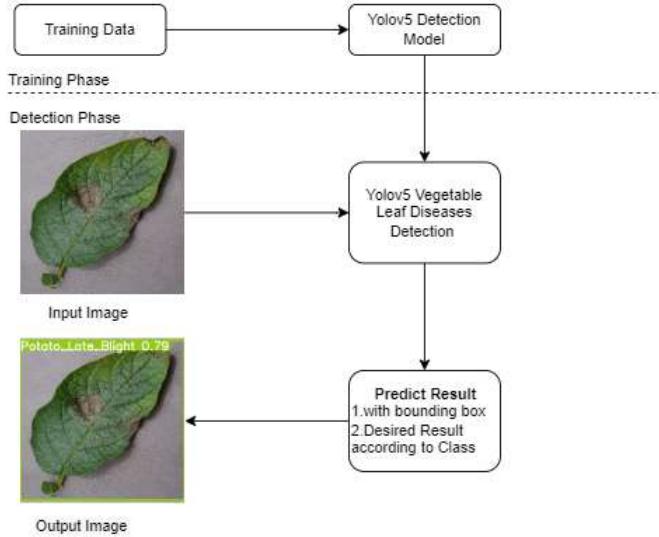


Fig. 5. Workflow of YOLOv5 Model for Plant Disease Prediction

into $n \times n$ cells, where $n > 3$. Each cell generates a vector with dimensions of $\#num_anchor_boxes \times (5 + m)$. Following that, a prediction system is given these features to create boxes around objects which include the data (pc, bx, by, bw, h, c1,...,cm). and determine their classes. pc represents the likelihood that an item falling into any of the categories c1,...,cm is contained within our anticipated bounding box and bx, by, bw, bh represents its dimensions.

It defines EfficientNet as the main backbone of Deep CNN, the feature networks BiFPN, and gives feedback with class/box prediction. This is a prediction tool for vegetable leaf disease that takes an input image, develops and feeds image characteristics through a prediction system, draws bounding boxes around the image, and predicts the classes.

This object detector combines the class labels from various networks with the bounding boxes. In Fig. 4, three components are illustrated which are necessary for YOLO Network: the backbone is CSP Darknet, the neck is for PANet, and the head is for YOLO. Before the data is sent to PANet for feature fusion, it is first passed to CSPDarknet for feature extraction. As followed in Fig. 5, the YOLOv5 output results per unit are Class, Score, Position, and Size.

3.5 Performed Tests

To evaluate the performances of the suggested model, we conducted several tests in various experimental configurations. The instructions given to the model changed several network parameters.

The complete dataset has been divided into 20% for validation and 80% for training. The dataset was then tested using the proposed mode. Some other models have been implemented using the pre-trained model, like VGG-16 and ResNet-50.

4 Result Analysis

In this Deep CNN model, the instruction was given to start the training with the training data set, which contained both the original and augmented images. Then a validation test was implemented to hypothesize the YOLOv5 model. The proposed network displays good convergence in the training phases, as seen in Table 1.

Table 1. Performance Matrices of the Proposed Classifier Based on Detecting Classes/Various Plant Diseases

Metrics	Corn_Blight_Disease	Corn_Common_Rust	Corn_Gray_Leaf_Spot_Disease	Corn_Healthy	Potato_Early_Blight	Potato_Healthy	Tomato_Late_Blightrotation	Potato_Late_Blight	Tomato_Early_Blightrotation	Tomato_Healthyrotation	Tomato_Mosaic_Virusrotation
Precision	0.823	0.964	0.891	0.989	0.998	0.985	0.895	0.995	0.785	0.857	0.983
Recall	0.664	0.86	0.887	1	1	1	0.887	1	0.833	0.802	0.785
Mean Average Precision	0.795	0.949	0.931	0.995	0.995	0.995	0.932	0.995	0.886	0.88	0.886

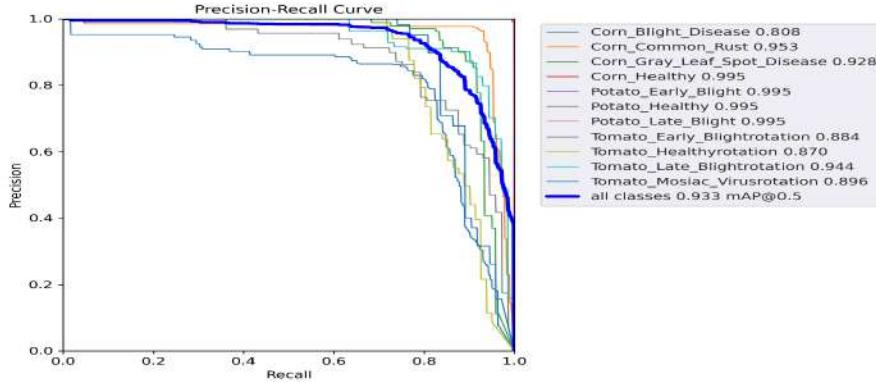


Fig. 6. Validation Testing on YOLOv5 Model (PrecisionRecall Curve)

Like other single-stage object detectors, YOLOv5 was developed with three components since it is a single-stage disease detector. The primary purpose of our proposed model Backbone is to extract unique characteristics from an input picture. To extract detailed, informative characteristics from an input picture in YOLOv5, the CSP (Cross Stage Partial Networks) are dedicated as the backbone.

Precision indicates how many positive classes the model properly predicted is positive. We divide the number of successfully classified positive cases by the number of anticipated positive examples to get the precision value. The equation is, $Precision = TP \div TP + FP$.

Recall measures how well the model foresaw all sorts of positive data. The recall rate is known as the proportion of all positively classified examples that

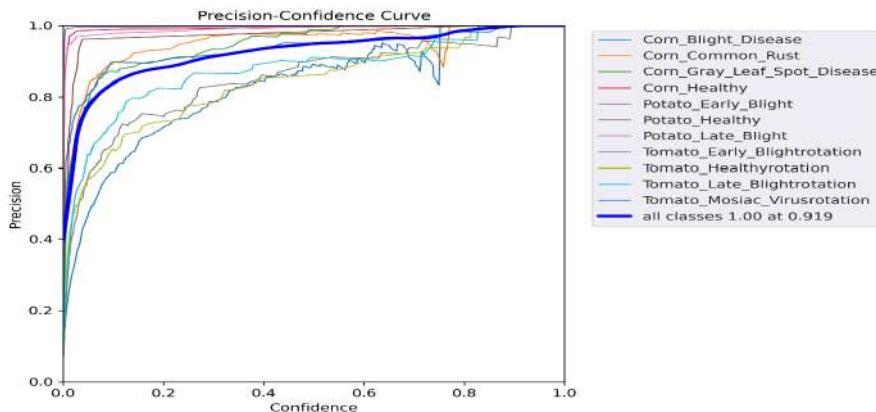


Fig. 7. Validation Testing on YOLOv5 Model (PrecisionConfidence Curve)

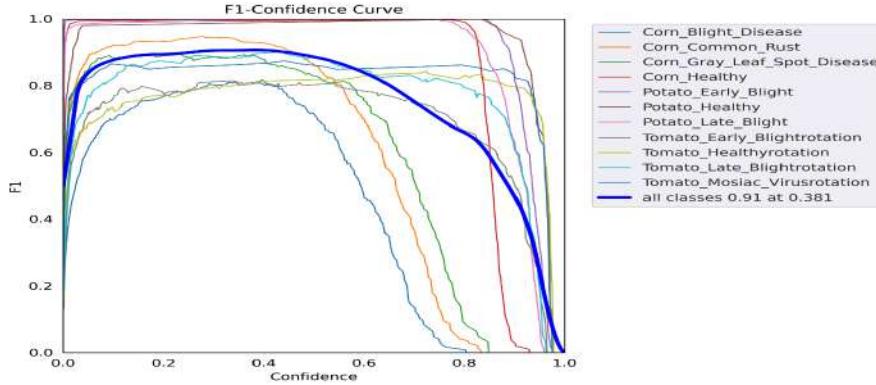


Fig. 8. Validation Testing on YOLOv5 model (f1-Confidence Curve)

were correctly classified to all positively classified instances. The equation is, $Recall = TP \div TP + FN$.

We get the proposed model using YOLOv5-based precision-recall results in Fig. 6. For 200 epochs with batch 40, precision gets 80% above accuracy and recall gets 90% above accuracy.

In Fig. 7, precision vs. confidence represents that for 200 epochs, every image in our vegetable leaf diseases dataset gets a mean confidence of 1.00 at 0.42 for every class. Still, precision gets 100% for the maximum class for our dataset. On the other hand, the YOLOv5-based f1 score vs. confidence curve is illustrated using Fig. 8.

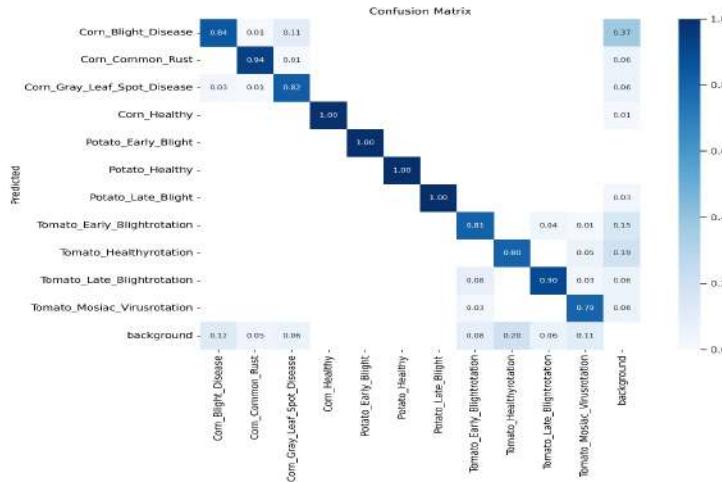


Fig. 9. Validation Testing on YOLOv5 Model (Confusion Matrix)

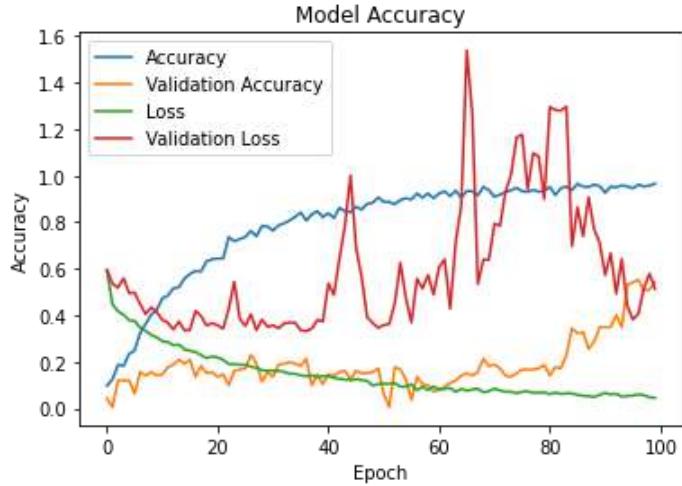


Fig. 10. Accuracy and Loss Curve (ResNet-50 Accuracy and Loss with Respect to Training and Validation Steps)

Comparative and transferable performance measures are necessary to determine how effective an algorithm or strategy is. Adopting training and testing sets, which might cause inconsistency in model performance, is the main challenge for evaluating any approach. The confusion matrix in Fig. 9, which includes true-positive (TP), true-negative (TN), false-positive (FP), and false-negative (FN) values, is the foundation for the majority of performance indicators. Depending on how the performance review is conducted, the importance of these components may change.

ResNet-50 makes it possible to train the network on thousands of layers without affecting performance. It is one of the several variations that can operate with 50 neural network layers. Fig. 10 represents ResNet-50-based accuracy against Epoch 100 for training accuracy, training loss, validation accuracy and validation loss.

VGG-16 is a convolutional neural network that is 16 layers deep and is considered the basis of ground-breaking object recognition models. VGG-16-based Accuracy against Epoch 100 for training and validation accuracy is represented in Fig. 11.

To compare the networks' output for the class of leaf diseases. To put it another way, both networks routinely offer accurate classifications. The categorization accuracy for each image across all categories may be seen by looking at all the figures. While VGG-16 predicts tomato_early_bright_rotation as the top prediction for the class, ResNet-50 essentially sees potato_late_blight disease as the top prediction. There is still a significant overlap between various categories for other, less common classifications.

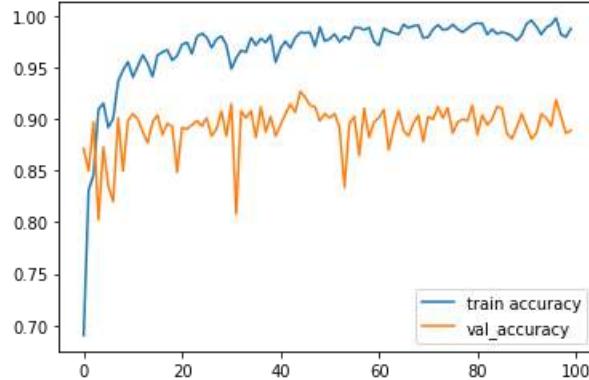


Fig. 11. Accuracy Curve (VGG-16 Accuracy of Training and Validation)

In Fig. 12, we can see the accuracy and loss of training & validation dataset of ResNet-50 & VGG-16, where we get an accuracy of 96.57% for ResNet-50 and 98.03% for VGG-16.

So, after implementing the VGG-16 and ResNet-50 pretrained model, we assumed that the VGG-16 model gained the highest accuracy rate. We also see that losses and convergence time are less than any other pre-trained models.

ResNet-50 has an overall 96% accuracy of detecting the right items in the dataset of vegetable leaf disease detection, according to a real-time investigation of convolutional neural network performance. The classification accuracy for VGG-16 and our suggested model, respectively, is 98% and 99%. CNN's performance on photos varies significantly from the outcomes of live testing. CNNs frequently misclassify a small number of objects during live testing; ResNet-50, for instance, frequently struggles to distinguish between dogs and deer. In

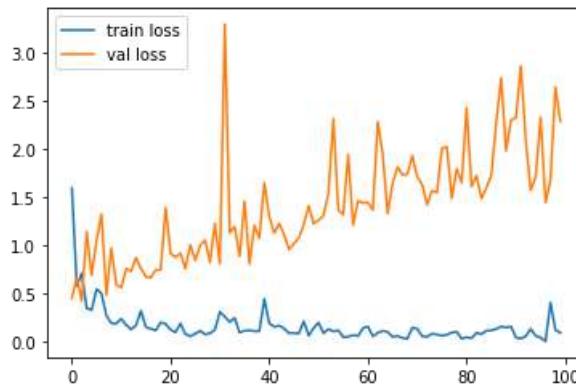


Fig. 12. Accuracy and Loss Curve (VGG-16 Loss of Training and Validation)

the majority of the scenes, it recognizes them as horses. The accuracy findings demonstrate that, in comparison to all other CNN algorithms, our model performs better and has the best detection accuracy.

5 Conclusion

In this study, we applied the theory of transfer learning and created an automated system to identify and categorize 11 illnesses in leaves. Our method can aid farmers in boosting crop yields and spotting illnesses in their early stages. YOLOv5's deep learning model is addressed in our work as the primary model, and ResNet-50 and VGG-16 are also discussed to compare the results with those of YOLOv5. To find 11 distinct types of leaf disease, all algorithms are used. A dataset of 5261 featuring images of healthy-unhealthy is used to train the YOLOv5 model. The trained model is tested on images of the diseases: tomato early blight rotation, tomato healthy rotation, tomato late blight rotation, and tomato mosaic varus rotation. The model has the highest accuracy, recall, and mAP and has a significantly higher recognition rate.

To improve production and the quality of the crop, it is crucial in future work to identify a disease in a plant when it is in the budding stage. Since disease diagnosis requires a great deal of skill, it would be beneficial if this system could be put in place on smartphones so that farmers could take a photo of a leaf and upload it to the server. The server will automatically recognize and categorize the ailment kind and transmit the results and recommended medications back to the smartphone. By implementing the suggested technique on a smart embedded system, this model may be utilized for real-time object identification to help farmers in the agricultural sector increase crop productivity.

References

1. Bhattacharai, S.: New plant diseases dataset (Nov 2018), <https://www.kaggle.com/datasets/vipooooool/new-plant-diseases-dataset>
2. Fenu, G., Malloci, F.M.: Diamos plant: A dataset for diagnosis and monitoring plant disease. *Agronomy* **11**(11), 2107 (2021)
3. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* **17**(9), 2022 (2017)
4. Ghazi, M.M., Yanikoglu, B., Aptoula, E.: Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing* **235**, 228–235 (2017)
5. Joshi, R.C., Kaushik, M., Dutta, M.K., Srivastava, A., Choudhary, N.: Virleafnet: Automatic analysis and viral disease diagnosis using deep-learning in vigna mungo plant. *Ecological Informatics* **61**, 101197 (2021)
6. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Communications of the ACM* **60**(6), 84–90 (2017)
7. Lee, S.H., Chan, C.S., Mayo, S.J., Remagnino, P.: How deep learning extracts and learns leaf features for plant classification. *Pattern Recognition* **71**, 1–13 (2017)

8. Lv, M., Zhou, G., He, M., Chen, A., Zhang, W., Hu, Y.: Maize leaf disease identification based on feature enhancement and dms-robust alexnet. *IEEE Access* **8**, 57952–57966 (2020). <https://doi.org/10.1109/ACCESS.2020.2982443>
9. Mukti, I.Z., Biswas, D.: Transfer learning based plant diseases detection using resnet50. In: 2019 4th International conference on electrical information and communication technology (EICT). pp. 1–6. IEEE (2019)
10. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014)
11. Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., Batra, N.: Plantdoc: a dataset for visual plant disease detection. In: Proceedings of the 7th ACM IKDD CoDS and 25th COMAD, pp. 249–253 (2020)
12. Sinha, A., Shekhawat, R.S.: Review of image processing approaches for detecting plant diseases. *IET Image Processing* **14**(8), 1427–1439 (2020)
13. Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D.: Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience* **2016** (2016)
14. Su, J., Liu, C., Coombes, M., Hu, X., Wang, C., Xu, X., Li, Q., Guo, L., Chen, W.H.: Wheat yellow rust monitoring by learning from multispectral uav aerial imagery. *Computers and electronics in agriculture* **155**, 157–166 (2018)
15. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
16. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10781–10790 (2020)
17. Thuan, D.: Evolution of yolo algorithm and yolov5: The state-of-the-art object detection algorithm (2021)
18. Tiwari, V., Joshi, R.C., Dutta, M.K.: Dense convolutional neural networks based multiclass plant disease detection and classification using leaf images. *Ecological Informatics* **63**, 101289 (2021)
19. Yadav, S., Sengar, N., Singh, A., Singh, A., Dutta, M.K.: Identification of disease using deep learning and evaluation of bacteriosis in peach leaf. *Ecological Informatics* **61**, 101247 (2021)
20. Zhang, X., Qiao, Y., Meng, F., Fan, C., Zhang, M.: Identification of maize leaf diseases using improved deep convolutional neural networks. *Ieee Access* **6**, 30370–30377 (2018)

Detection of Covid-19 using an Infrared Fever Screening System (IFSS) based on Deep Learning Technology

V Muthu¹ and S Kavitha²

¹Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

²Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

¹muthutech77@gmail.com, ²kavithas@srmist.edu.in

* V Muthu and S Kavitha

Abstract. Treatment for the new Coronavirus is both expensive and challenging, preventing infections may have a significant impact on healthcare costs and quality of life. This research presents a novel paradigm for identifying people displaying symptoms of covid-19 infection in public settings, one that makes use of Convolution Neural Network (CNN) and deep machine learning methods. Infection with the COVID-19 virus is characterized by a high temperature, or fever. Accordingly, the goal of this research is to develop a prototype for an automated system that can detect and isolate covid-19 carriers in public settings. The study's suggested prototype is novel and cutting-edge in many ways: Our system consists of three components: 1.An Infrared Fever Screening System that automatically detects thermal signals and measures temperature to check whether an individual has a fever or not; 2.HD visual/facial auto-calibration system that provides accurate and presides facial landmarks that help track people and measure their temperature at various regions; 3.A real-time sensor fusion of visual and thermal camera data that forms a single model with multiple lidars, racial recognition. The proposed prototype accurately detects people with temperatures above average, even when an individual has a face mask. It balances the strengths of different sensors that can detect human body temperature from facial landmarks such as the forehead and the eyes..

Keywords: Convolutional Neural Network, Covid-19, DeepLearning, Face Detection, Fever Screening, Image Processing, IR Sensor, Thermal Imaging.

1 Introduction

Since the outbreak of COVID-19 in 2019 and its spreading into a pandemic, different governments in different countries have been putting countermeasures in place, such as quarantine to curb the transmission of this disease. Since fever is a key sign and symptom of this disease, many temperature sensors and detection

systems have been developed to help identify people with this disease and isolate them from the uninfected population. Most devices today use non-contact temperature sensors to detect and measure human temperature to reduce the risk of transmitting Coronavirus through contact. Nonetheless, technology is also advancing rapidly, and new and more advanced devices and systems are developed daily. These devices play a critical role in detecting and controlling the spread of the current COVID-19 pandemic.

This article presents an Infrared fever screening system (IFSS) that screens people with fever in real-time as they do their business in an airport check point. The proposed prototype uses motion and facial detection technology to automatically track each individual who enters an airport and screen them for fever [6]. The system uses automated motion and facial sensor to detect and screen each individual getting in to the airport checkpoint. This eliminates the need for an individual having to stop for their body temperature to be checked or place their hand near a non-contact temperature detector. It allows free movement of people in an airport and reduces overcrowding at the checkpoint. It also makes it easier to screen traveler's and reduces the risk of transmitting the Coronavirus among travelers [6]. The system uses infrared thermography and thermal image processing to capture human facial images and measure their temperature using the infrared radiation emitted from different facial regions, such as the forehead and the inner canthus of the eyes. Fig. 1 below shows a checkpoint where people need to manually check their temperature and another checkpoint where people can move freely in an airport checkpoint as their temperature is measured automatically while maintaining social distance.



Fig. 1. Manual temperature detection (Left) VS Automated infrared temperature detection (Right)

The proposed automated infrared fever screening system will allow staff at the airport checkpoint to measure the temperature of their client from a safe distance and eliminate the need for human intervention. At airport checkpoints, staff deal

with multiple people, bringing a major concern for COVID-19 since the disease can easily spread from one person to another when in close contact[11][14]. IFSS eliminates this problem by taking advantage of the security camera at the airport checkpoint to perform sensor fusion of thermal and visual data in real-time and in a free movement of travellers in and out of the airport. The system is also connected to cloud servers that collect thermal data, analytics processes, and future references. IFSS is an advanced system based on deep machine learning technology. It offers different benefits and advantages in the process of curbing the transmission of Coronavirus through air travel by:

1. Provides a novel screening solution that uses simultaneous techniques to measure the temperature of multiple travellers in real-time from a distance without human intervention.
2. Providing a novel sensor fusion technique fuses thermal and visual frames/images to detect and measure travellers' temperature at various distances, and even then, travellers have hats, sunglasses, and masks.
3. Providing a novel screening solution that is safer for travellers and airport staff at the airport checkpoint reduces the need to come into contact during the temperature screening process.

2. Background and challenges

Infrared Fever Screening System (IFSS) is one of the most advanced systems in the market today, which allows huge organizations with a high traffic flow of people in and out of the premises to measure temperature and warn when an individual shows abnormal temperature. The system is designed to identify and locate an individual with a temperature above a predetermined threshold. According to the Center for Disease Control and Prevention, an individual with a body temperature above 38° C or 100.4° F is considered to have a fever. The normal skin temperature for a human being is about 33° C or 91° F.

Although the body temperature is not equally distributed, it is possible to measure the average body temperature from different facial landmarks such as the eyes using visual and thermal imaging and processing technology. Measuring the human body temperature through facial screening thermography is one of the most researched topics today. Most research studies reveal that it is an accurate and effective way to detect human fever and diseases. Fig. 2 show shows how an individual can use this technology to detect and measure human body temperature from different facial landmarks.

Nonetheless, there are several challenges that IFSS is still yet to address to achieve its capability to detect and measure human body temperature automatically and in real-time. First, the reliability of measuring the core temperature of many people simultaneously can be limited since people in a crowded place can occlude each other [15]. Secondly, people can have to obstruct objects on their faces, such as masks, glasses, and caps, which can interrupt the input of thermal and visual images used to measure human body temperature.

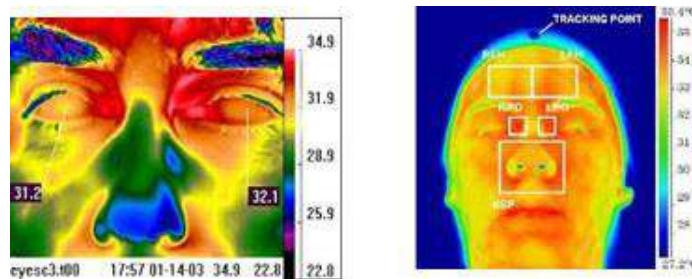


Fig.2. A Thermal image with different temperatures in different facial landmarks

2.1 Related work and research gap

Due to the unprecedented COVID-19 pandemic, many researchers and scientists have become very interested in finding ways to help the government, organizations, and individuals adopt and mitigate the risks of getting infected or spreading the virus. Many studies show that most of the challenges and problems related to COVID-19 are still not well addressed, and this disease is a problem that will continue to impact business and human activities now and in the future [4][7]. The transportation industry, such as the airport, is most affected by these outbreaks and continuing pandemics. Most airports use temperature detectors that need human intervention to screen travellers for potential COVI-19 infection. The approach allows staff at the checkpoint to accurately detect and identify people with abnormal body temperature or fever and isolate them from other travellers to reduce the risk of them spreading COVID-19 if infected.

However, this approach has also been proven impractical due to the high traffic situation at most airports' checkpoints. Due to this problem, many researchers have proposed and recommended using contactless infrared fever screening systems integrated with visual and thermal cameras. These systems are also integrated with data collection applications that produce a dataset from which a machine can learn to detect and identify people with fever and signs of COVID-19 infection [2][4]. Traditionally, thermal-only and optical-thermal detectors can use infrared images to detect facial features such as eye canthus and measure human body temperature. Machine learning technology is also critical in providing facial algorithms that help machines learn and use statistical models to analyze and draw inferences from infrared images. Through this technology, machines are becoming more accurate in analysing facial infrared images and predicting input data output. [9] reveal that through open-source datasets, individuals can use machine learning models to train machines to detect visual images such as facial landmarks and automatically screen people for COVID-19 in public places.

Although machine learning technology has paved the way for using infrared imaging and fever screening, making it easier to identify people with COVID-19 in places with high traffic flow, its application has some major limitations. For

instance, an IFSS model based on machine learning has poor transfer learning ability and limited reusability of the modules. The system relies entirely on human input to learn, so its performance cannot be guaranteed for consistent accuracy and precision, especially in a dynamic environment. In general, IFSS based on machine learning technology and requires a lot of human input and supervision to learn, which makes it impractical for real-time detection of people infected with COVID-19 in an environment with a fast traffic flow of people, such as an airport checkpoint. In contrast, the deep learning model uses artificial neural networks with multiple layers of processing data. It allows the machine to automatically adopt and develop new features for analysing infrared images to identify individuals infected with COVID-19 by measuring human body temperature.

[10] Deep learning technology is good for developing practical and affordable modalities and techniques for COVID-19 diagnosis. He also suggests that through artificial intelligence, an individual enhances image character and improves the accuracy and capability of the machine in anticipating the COVID-19 virus. He studies how different deep learning models are. He reveals that Deep learning techniques that use conventional neural networks to train and detect COVID-19 outperform traditional techniques that use other methods such as machine learning.[5] Suggest that deep learning techniques can assist computers in analysing lungs ultrasound imagery and enable screening tools to diagnose people for COVID-19 virus. The researchers in this study reveal that a screening system based on deep learning can detect the COVID-19 virus using ultrasound data with a significantly high level of accuracy and consistency. [3] My proposed an automated deep learning-based classification model based on Convolutional Neural Network (CNN)[1] and conducted a study to test its efficiency. The result of the study shows that the proposed model produced the highest accuracy in classification and identification of the symptoms of COVID-19 from X-ray images than other existing models. All these studies show a screening system based on deep learning technology. CNN has a high potential to accurately classify and identify a symptom of COVID-19 from X-ray and CTR images/pictures. However, none of this study focuses on how a screening system based on deep learning technology and CNN can use facial thermal images to identify symptoms and detect COVID-19.

2.2 Overview of the proposed IFSS based on the Deep learning model

The proposed Infrared fever screening system in this article is designed to operate in a free-flow manner where people are not required to stop or pause as they enter the airport checkpoint for their temperature to be checked. It also does not require the staff at the check port to contact the travellers to ensure they measure their temperature. Instead, it has a motion sensor that detects humans moving in and out of the checkpoint section and a thermal sensor that automatically measures the temperature of each person at the checkpoint and provides their temperature on a screen. The major aspect of this system is that it is countless and hence increases the safety of the staff at the checkpoint. It also works in real-time and can measure the temperature of many individuals simultaneously, allowing an airport to save

time at the checkpoint.

Figure 3 shows a setup of the deployment of the IFSS system at an airport checkpoint. The image on the left shows how the IFSS system can be integrated into the CCTV security system to form a 2in1 system that can measure the temperature of travellers automatically while recording the camera security footage. The camera is located at a designated point where it can detect their motion, capture their facial image and measure their temperature automatically when people enter the entrance. It also has wide fields of view, allowing the system to capture many people simultaneously, even in motion [15]. After capturing and processing the visual and thermal images, the system displays the data on the screen, where an operator can evaluate and monitor each individual. Suppose the system detects an individual among the crowd has a fever or abnormal temperature. In that case, it identifies that individual through a facial recognition sensor which helps the staff monitor the screen to isolate them from the crowd for further assessment.

The proposed IFSS work similarly to the CCTV system, where people suspected of having COVID-19 are requested to step aside where the final assessment is done by a healthcare professional with proper personal protective equipment. This makes it easier to control and maintain physical contact between the checkpoint staff and travellers since the screening process is done automatically and with minimum need for close contact [15]. In addition, the entire process can be controlled from a central room where the operator at the checkpoint can request an individual to step aside from the crowd using audio speakers without coming in contact with the travellers.

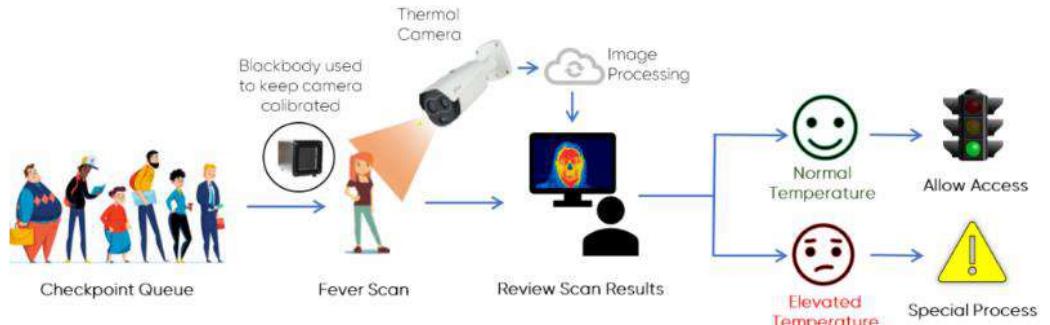


Fig. 3. IFSS Deployment

3. IFSS Convolution Neural Network (CNN) design

The proposed IFSS system in this study is based on CNN, which forms the entire model's primary architecture components. The system consists of multiple layers with different functions that classify the visual and thermal images as input data and then process it to provide the screening data for each individual as output [12].

3.1 Convolutional

The system extracts feature from the input images through a convolution process that involves filtering and feature mapping to learn and produce output data. It has a kernel that uses the matrix number processes over the input images to transform images into a feature map [12].

The subsequent feature maps are computed through $O[m,n]=(1*F)[m,n]=\sum F[i,j]I[m+i, n+j]$ as the general formula in this formula I denotes the Input Image. F denotes the kernel while the indexes (m,n) are the rows and column, respectively, and (m*n) denote the dimension or pixel of the filters [13]. The convolution process also uses a nonlinear operation to eliminate negative pixel values based on the ReLu activation function.

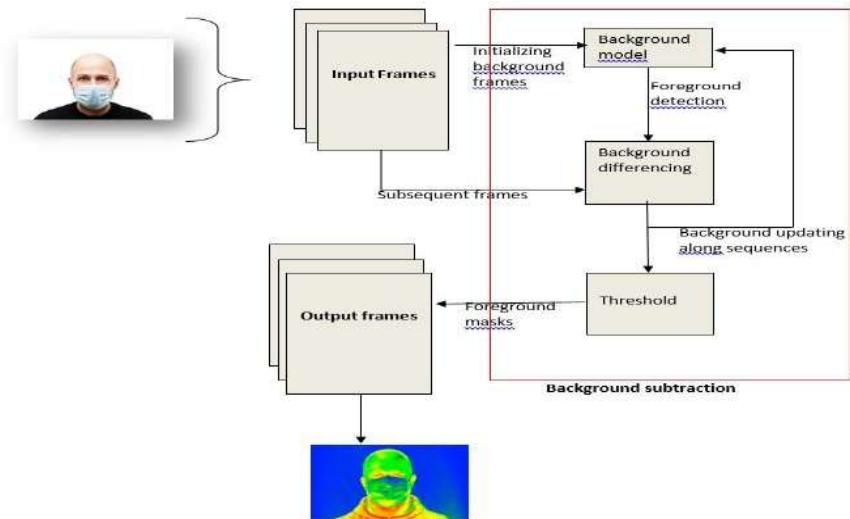


Fig. 4. Schematic diagram of background subtraction

3.2 Pooling layers

Generally, a pooling layer is a subsequent layer after the convolutional layer that reduces the feature maps' dimensions, allowing the machine to automatically learn different computations in the network. The proposed IFSS system in this study uses this layer after the nonlinear ReLu activation function to summarize the features present in the region of the feature map generated by the convolution layer [16].

There are three types of pooling layers: max pooling, average pooling, and global pooling. Max pooling is a pooling layer and an operation in CNN that selects the maximum element from the region of the feature map in a filter. This operation allows a CNN-based system to produce a feature map that contains prominent features of the feature map from the convolutional layer. Unlike the max pooling operation, the average pooling operation provides the average feature present in a respective patch. The global pooling operation, on the other hand, reduces each operation in a CNN-based system to a single value.

3.3 Fully connected layer

The output from the pooling layer becomes the input for the full connected layer, which forms a feed-forward network and the last pooling layer of the convolutional layer. IT is a three-dimensional matrix based on an artificial neural network that performs the same mathematical operations. In the proposed IFSS system, the artificial neural network computes the $g(Wx+b)$ calculation in each layer. In this calculation, X is the input vector with dimension $[p_{-1}, 1]$, and W is the weight matrix with dimensions $[p_1, n_1]$, where p_1 is the value of neurons in the previous layer and n_1 is the value of neuron in the current layer [8]. The initial b is the bias vector with dimension $[p_1, 1]$, and g is the activation function base on the ReLu activation function. This calculation helps the system to predict the classification of the visual and thermal images and produce based on the interconnected neurons of the feature map [8]. The system also uses extensive thermal data images, which form the pre-trained network that the system learns from, and mask the features in the input patch to detect human temperature based on the thermal video frames. In general, the above-stated work provides a significant result which depicts that the proposed system in this study has a chance to provide reliable and valid results. Fig. 5 shows a schematic diagram that outlines CNN's building block.

4. Experimental setup, methodology, and result

The proposed system uses visual and thermal data as the input frames fed to the CNN for feature mapping, extraction, and poling process processes. The study also introduces a traditional handheld contact thermometer with manual measurement. It then collects the result of each system to evaluate the effectiveness of the proposed system compared to the existing system in most airports today.

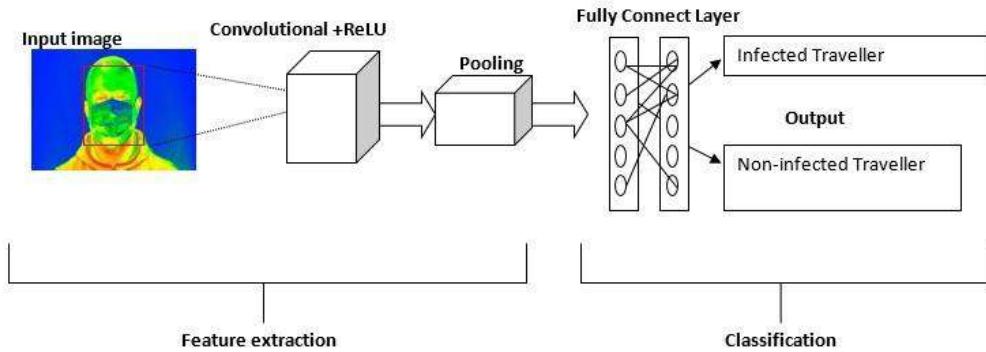


Fig. 5. Schematic Diagram of CNN

The researcher in this study assumes that since the proposed system uses advanced technology, it will be more accurate and highly efficient in detecting traveller's with COVID-19 [7]. For experimental purposes, the study has two datasets, PRODUCTION A and PRODUCTION B, as shown in Table 1 below. All the two datasets are in the same lab and the same environment. The same entrance is used for the screening process, and at the same time way two service desks are provided for the screening process.

Table 1: Dataset used in the experiment

Dataset	Number of traveller's screened with negative results	Number of traveller's Screened with Positive results	Total	Time Take for the screening process
Production A	270	20	290	12 hours
Production B	520	50	578	12 hours

One desk uses the traditional handheld contact thermometer, and the other uses the Infrared fever screening and contactless system. The total number of people screened, both those with a fever and those without a fever, is recorded as shown in Table one below. The traditional handheld contact thermometer uses the PRODUCTION A dataset, while the Infrared Fever Screening and contactless system use the PRODUCTION B dataset. Both systems use a thermal sensor for the screening process, and we assume the result of each system is accurate.

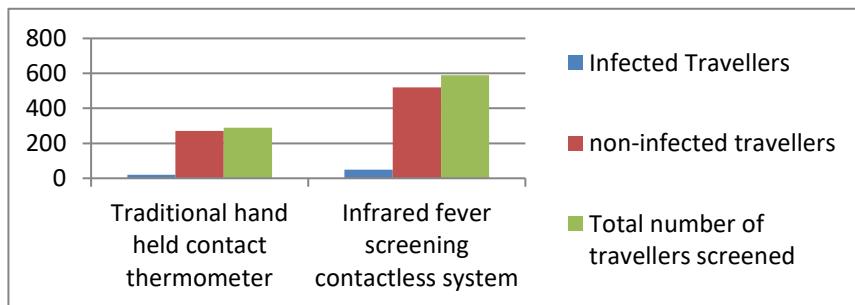


Fig. 6. Efficiency of the traditional-hand-held contact thermometer VS Infrared Screening contactless screening system

Comparative analysis

When the research compared the proposed Infrared Fever Screening System (IFSS) against the Traditional-handheld-contact-thermometer (the Traditional-handheld-contact-thermometer is identified as the existing system in this case) the results were as shown in Table 2 below. This analysis reveals that the ISS system was better than the traditional-handheld-contact-thermometer in terms of application, speed and effectiveness. Unlike the existing model the proposed model is automated which eliminates the need for physical human intervention. The deep learning technology, embedded in the system also allow it to perform computations processes faster and more effectively than the existing system.

Table 2: Infrared Fever Screening System verse Traditional-handheld-contact-thermometer

Comparative feature	Traditional-handheld-contact-thermometer	Infrared Fever Screening System (IFSS)
Application	Manual	Automated
Accuracy	Moderate	Moderate
Speed	Low	High
Effectiveness	Low	High

5. Discussion

Research studies have been carried out to support the perspective that the proposed model can be implemented in places with a high traffic flow of people, like airport checkpoints. The result of the deployed experimental system reveals that implementing the proposed infrared fever screening system in an airport checkpoint can reduce the risk of spreading COVID-19 by detecting infected travellers and isolating them from no-infected travellers [4]. It also can provide a safer system for the airport checkpoint to screen travellers for COVID-19. The proposed system can detect COVID-19 using a thermal sensor more effectively than the traditional handheld thermometer with a thermal sensor.

The CNN in the proposed IFSS system, is based on multiclass layers with subsequent feature maps and mathematical algorithms achieved through visual and thermal sensors to measure human temperature and detect people infected with Coronavirus automatically. The trained network automatically screens fever in real-time by displaying the human body temperature based on the predetermined threshold and alerting the staff at the checkpoint of individuals suspected of being infected with the Coronavirus. It combines the security system with the thermal camera hence provide better uses the resources at the airport checkpoint. The experiments also show that one individual can perform security and screening roles simultaneously using the same screen display.

6. Conclusion

The purpose of this study was to conduct an experimental study to determine the effectiveness of the proposed Infrared Fever Screening System (IFSS) as compared to the traditional handheld-thermometer. The result reveals that proposed model, Infrared Fever Screening System (IFSS), is a more effective system for has the potential to screen human body temperature automatically and simultaneously. It requires less human intervention and allow check point officer to carry out their tasks at a safer instance which reduce risk of infection. The automated technology in the propose system also increase the speed at which travellers get screened at the airport checkpoint which reduce the risk of overcrowding by allow individuals to move freely without the need of queuing and waiting to screened. The IFSS model also has moderate rate of accuracy which although not high, is significant given that IFSS model has more advantages over the traditional handheld-thermometer.

References

1. Aishwarya, T., & Ravi Kumar, V. (2021). Machine learning and deep learning approaches to analyze and detect COVID-19: a review. *SN computer science*, 2(3), 1-9.
2. Ahn, S. (2016). Deep learning architectures and applications. *Journal of Intelligence and Information Systems*, 22(2), 127-142.
3. Akter, S., Shamrat, F. J. M., Chakraborty, S., Karim, A., & Azam, S. (2021). COVID-19 detection using deep learning algorithm on chest X-ray images. *Biology*, 10(11), 1174.
4. Brzezinski, R. Y., Rabin, N., Lewis, N., Peled, R., Kerpel, A., Tsur, A. M., ... & Hoffer, O. (2021). Automated processing of thermal imaging to detect COVID-19. *Scientific Reports*, 11(1), 1-10.
5. [5] Diaz-Escobar, J., Ordóñez-Guillén, N. E., Villarreal-Reyes, S., Galaviz-Mosqueda, A., Kober, V., Rivera-Rodriguez, R., & Rizk, J. E. L. (2021). Deep-learning-based detection of COVID-19 using lung ultrasound imagery. *Plos one*, 16(8), e0255886.
6. Ghassemi, P., Pfefer, T. J., Casamento, J. P., Simpson, R., & Wang, Q. (2018). Best practices for standardized performance testing of infrared thermographs intended for fever screening. *PloS one*, 13(9), e0203302.
7. Grewe, L., Choudhary, S., Gallegos, E., Jain, D. P., & Aguilera, P. (2021, May). Low-resolution infrared temperature analysis for disease situation awareness via machine learning on a mobile platform. *Signal Processing, Sensor/Information Fusion, and Target Recognition XXX* (Vol. 11756, pp. 287–299). SPIE.
8. Huang, C., Fan, J., Li, W., Chen, X., & Zhu, Q. (2019). Reach Reachability analysis of neural-network controlled systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 18(5s), pp. 1–22.
9. Katte, P., Kakileti, S. T., Madhu, H. J., & Manjunath, G. (2022). Automated Thermal Screening for COVID-19 using Machine Learning. *arXiv preprint arXiv:2203.14128*.
10. Kumar, M., Atalla, S., Almuraqab, N., & Moonesar, I. A. (2022). Detection of COVID-19 Using Deep Learning Techniques and Cost Effectiveness Evaluation: A Survey. *Frontiers in Artificial Intelligence*, 5.
11. Martinez-Jimenez, M. A., Loza-Gonzalez, V. M., Kolosovas-Machuca, E. S., Yanes-Lane, M. E., Ramirez-GarciaLuna, A. S., & Ramirez-GarciaLuna, J. L. (2021). Diagnostic accuracy of infrared thermal imaging for detecting COVID-19 infection in minimally symptomatic patients. *European journal of clinical investigation*, 51(3), e13474.
12. Müller, D., Ehlen, A., & Valeske, B. (2021). Convolutional neural networks for semantic segmentation as a tool for multiclass face analysis in thermal infrared. *Journal of nondestructive evaluation*, 40(1), 1-10.
13. Peddinti, B., Shaikh, A., Bhavya, K. R., & KC, N. K. (2021). Framework for Real-Time Detection and Identification of possible patients of COVID-19 in public places. *Biomedical Signal Processing and Control*, p. 68, 102605.
14. Pool, R. (2022). Flying in the COVID-19 Era: Science-based Risk Assessments and Mitigation Strategies on the Ground and Air: Proceedings of a Workshop.

15. Tan, Y. H., Teo, C. W., Ong, E., Tan, L. B., & Soo, M. J. (2004, April). Development and deployment of infrared fever screening systems. In Thermosense XXVI (Vol. 5405, pp. 68-78). SPIE.
16. Zulkifley, M. A., Abdani, S. R., & Zulkifley, N. H. (2020). COVID-19 screening using a lightweight convolutional neural network with generative adversarial network data augmentation. *Symmetry*, 12(9), 1530.

Improving the power quality of wind turbines under unbalance voltage conditions using the SMC approach.

*HARIPRASAD B¹,Dr.Sreenivasan G²,Mahesh Kumar D³, and Lakshmi Swapna T⁴

¹Assistant Professor , Department of EEE, P.V.K.K.I.T, Anathapur, A.P. India
^{1*}vanoorhari@gmail.com

² Professor , Department of EEE, P.V.K.K.I.T, Anathapur, A.P. India
²gsn.anusree@gmail.com

¹Assosiate Professor , Department of EEE, P.V.K.K.I.T, Anathapur, A.P. India
^{3*} itsdmahesh@gmail.com

¹PG Student , Department of EEE, P.V.K.K.I.T, Anathapur, A.P. India
^{4*} tlakshmiswapna@gmail.com

Abstract. At the moment, one of wind energy's most challenging problems is grid integration. The technical specifications for wind turbines vary, and they must be connected to networks with various voltage disturbances, such as voltage imbalances and harmonics. Additionally, the standards have a limit on how much total harmonic current the wind turbine can introduce into the grid. Variable speed turbines frequently use doubly fed induction generators (DFIG). However, one issue with DFIGs is that they are very sensitive to changes in grid voltage. Wind turbine output quality can be decreased and DFIG parameters can change as a result of even slight changes in grid voltage, which can result in significant increases in stator and rotor currents. Wind turbines have historically been shielded from voltage fluctuations using a variety of techniques. Stem channels are one of the most popular techniques. However, DFIGs consume a significant amount of reactive power during grid outages, and the resulting blocking circuitry exacerbates the issue. Numerous control strategies have been put forth in compiled works to successfully control the grid side (GSC) , rotor side controllers (RSC) of DFIGs and improve the power quality.

Keywords: Wind Turbine, Power quality, Unbalanced Voltage, Integral terminal sliding mode control.

1 Introduction

Wind is a popular renewable energy resource of power that has lesser impact on environment compared to burning of fossil fuels. But the wind energy penetration into the grid network is the main problem with this wind energy. Though there are many disturbances like harmonics and unbalances, the wind turbines are needed to cooperate with technical essentials during such conditions to suppress the bad effects and remain connected to the grid. Furthermore, the harmonic current injected by wind turbine into the grid is limited by the standard value[1-3].

Generally, in variable speed wind turbines, DFIG are used. The problem with DFIG's is that they are sensitive to voltage fluctuations, for instance though there are slight grid voltage variations cause increase in rotor and stator currents which lead to damage of converters of DFIG and the power quality of wind turbine output power worsens. For the protection of the wind turbines from variations of voltage, several approaches were proposed and used. Among several approaches crowbar circuits are most commonly used. But these circuits rise the problem of absorption of reactive power during faults. To reduce the voltage sag problems Series dynamic resistors (SDR) were used . But SDR reduces the ability of DFIG's reactive power supply to grid. Though dynamic voltage restorers (DVR) and series grid side converters (SGSC) are able to mitigate voltage unbalances, these requires extra energy storage equipment which increases the capital outlay. To control DFIGs the grid side converters (GSC) and rotor side converters (RSC) and to reduce voltage unbalances of grid several approaches are introduced. Generally normal PI controllers are used to mitigate sag and to control RSC, GSC. But these PI controllers cannot provide robustness to parametric variation and external disturbances.

Sliding mode control (SMC of first order for DFIG wind turbine(WT) was developed in. To DFIG wind generator connected to the micro-grid for regulation of voltage a sliding mode controller is explained in. But this standard SMC high-frequency switching phenomenon, that in return is capable of exciting unmodeled dynamics of system, overheating the DFIG and lead to mechanical parts damage. Conventional SMC convergence time of system dynamics is not finite. Integral terminal sliding mode control(TSMC) approach was designed for DFIG grid side converter. But TSMC is not effective because of singularity problem. To overcome this singularity problem ITSMC approaches are proposed instead. For tuning of parameters of ITSMC controller automatically so as to achieve self-adaption a fuzzy controller is used Your contribution should be prepared in Microsoft Word.

2 DFIG Based Wind Turbine Modeling

DFIGs are mainly employed in the variable speed wind turbine applications. DFIG based wind turbine(WT) connected to microgrid is represented diagrammatically as shown in fig.1. Here RSC and GSC are connected with a coupling capacitor in between them. The RSC and GSC are done separately with different controllers each.

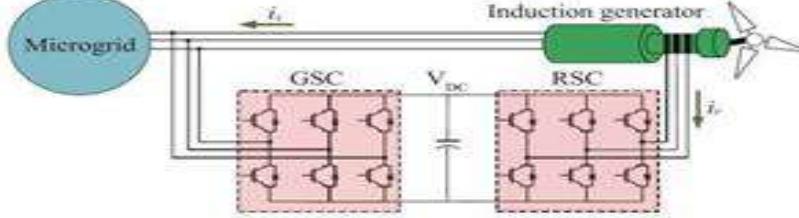


Fig 1. DFIG based wind turbine schematic diagram

2.1. RSC modelling

The state space model representation is made by assuming the system as systematic and balanced one in a synchronously rotating frame. DFIG of the state space model is rotating in d – q frame [4] is given by:

$$\dot{X} = f(x, t) + g(x, t)u \quad \dots\dots\dots(1)$$

Where,

$$f(x, t) = \begin{bmatrix} -\frac{R_r}{(\sigma L_r)^2} (\sigma L_r i_{r,d} + \frac{MV_{s,q}}{\omega_s L_s}) + \frac{R_r M \varphi_{s,d}}{L_s (\sigma L_r)^2} + \frac{\omega_r}{\sigma L_s} (\sigma L_s i_{r,q} + \frac{MV_{s,d}}{\omega_s L_s}) \\ -\frac{R_r}{(\sigma L_r)^2} (\sigma L_r i_{r,q} + \frac{MV_{s,d}}{\omega_s L_s}) + \frac{R_r M \varphi_{s,q}}{L_s (\sigma L_r)^2} + \frac{\omega_r}{\sigma L_s} (\sigma L_s i_{r,d} + \frac{MV_{s,q}}{\omega_s L_s}) \end{bmatrix}$$

And

$$g(x, t) = \begin{bmatrix} \frac{1}{\sigma L_r} & 0 \\ 0 & \frac{1}{\sigma L_r} \end{bmatrix}$$

here L_r and R_r rotor inductance and resistance, L_s is inductance of stator, M is mutual inductance, ω_s is synchronous speed, i_{rd} and i_{rq} are rotor currents in d,q axis respectively, $\varphi_{s,d}$ and $\varphi_{s,q}$ are stator flux in d,q axis respectively, $V_{s,d}$ and $V_{s,q}$ are stator voltage in d, q axis respectively.

2.2. GSC modelling

GSCs are generally involved in maintaining the DFIGs of DC-link voltage and minimizing the total reactive power and active power oscillations and supplied to the grid. Hence because of above mentioned reasons there should be proper GSC controlling in order to provide fixed DC voltage and improve quality of wind turbine power output under grid unbalanced conditions. The modeling of GSC in decoupling scheme is as follows.

$$\frac{di_{g,q}}{dt} = \frac{1}{L_g} (V_{s,q} - R_g i_{g,q} + \omega_s L_g i_{g,d}) - \frac{1}{L_g} V_{g,q} \quad \dots\dots\dots(2)$$

Where, $i_{g,d}$ and $i_{g,q}$ are grid current parts in d, q axis respectively, R_g , L_g are GSC resistance and inductance, $V_{g,d}$ and $V_{g,q}$ are grid voltage components in d, q axis respectively, ω_s is synchronous speed of DFIG.

2.3. ITSMC design for GSC and RSC

ITSMC design for GSC and RSC The grid and rotor side currents are transformed into d-q frame as d-q components of currents $s_{k,j}$. Now the tracking errors of the d,q components of the grid and rotor side currents $s_{k,j}$ are calculated as follows;

$$\dot{s}_{k,j}(t) = \dot{i}_{k,j\text{-ref}} - \dot{i}_{k,j} \quad \dots \quad (3)$$

where $S_{k,j\text{-ref}}$, $j = d, q$; $k = r, g$ are the reference values of d,q components for the rotor side current and grid side currents. Since the operation of converter is in d, q frame[4]model. Therefore for this reason the stator and rotor currents are divided into real power part which is called d component and a reactive power part which is called q component. The difference between actual value of real power of generator and reference value set by speed of wind is given to the controller. This difference value is used to produce $\dot{i}_{r,d\text{-ref}}$ i.e, d component of rotor current reference value . In the same way, the difference between actual value of reactive power of generator and reference value (generally zero) is given to the controller. This difference value is used to produce $\dot{i}_{r,q\text{-ref}}$ i.e, q component of rotor current reference value. The rotor current $S_{k,j}$ are given as follows.

$$S_{k,j} = s_{k,j} + \beta_{1,k,j} e_{1,k,j} s_{k,j} + \beta_{2,k,j} (s_{k,j}^2)^{-\alpha_k} S_{k,j} e^{-\lambda_k t} \quad \dots \quad (4)$$

where α_k, λ_k , $\beta_{1,k,j}$ and $\beta_{2,k,j}$ are positive constants.

Fuzzy Logic Controller.

Fuzzy logic is multivalued logic that logically allows for any real number between 0 and 1, inclusive, as the true value of each variable. The idea of partial truth, where the truth value can range from completely true to completely false, is accommodated by it. In contrast, in Boolean logic, variables can only have one of two possible integer values 0 or 1. Lotfi Zadeh coined the phrase fuzzy logic in 1998[5]. Fuzzy logic, but primarily Lukasiewicz and Tarski since the 1920s as Finite Value Logic. False reasoning is based on the idea that people tend to base their decisions on vague, non-statistical information. A mathematical method for representing ambiguous and imprecise data is to use fuzzy models or sets. These models can recognize, represent, control, and interpret.

Classical logic only allows true or false conclusions. However, there are also ideas with variable responses, such as when a group of people is asked to identify a color. In such cases, when the sample responses are plotted on a spectrum, the truth is the result of a judgment based on inaccurate or partial knowledge[6]. Both the truth level and the probability range from 0 to 1.

The base program can reference various dynamic components. For example, a temperature gauge for antilock brakes may have several different member functions that specify the specific temperature required to operate the brakes correctly. The same temperature value is set by each function to a truth value between 0 and 1. To decide how to manage the fruit, one can use this truth value. A method for expressing uncertainty is offered by fuzzy set theory.

Fuzzy logic works utilizing fuzzy set theory, in which a variable is individual from at least one sets. Every variable is characterized with a degree of membership. This Fuzzy logic licenses us to imitate the human psyche thinking measure in PCs, uncertain data evaluation , settle on choice dependent on dubious and deficient

information, yet by applying a "defuzzification" measure, come to clear end results.

The FLC mainly consists of three blocks.

Fuzzification
Inference mechanism
Defuzzification

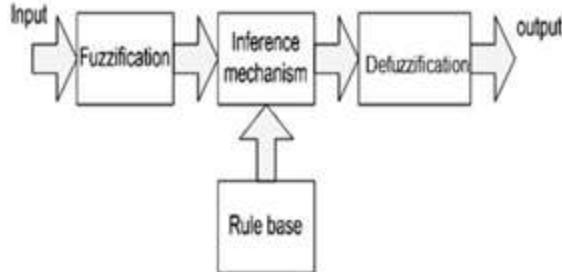


Fig 2: Block diagram of fuzzy logic controller.

Fuzzification

There are two ways in changing a crisp set into fuzzy set.

Support Fuzzification method:

The equation shown below,

$$\tilde{A} = \mu_1 q(x_1) + \mu_2 q(x_2) + \dots + \mu_n q(x_n) \quad (4)$$

Here $q(x_i)$ is part of fuzzification

fluffy set $q(x_i)$.

Grade Fuzzification method:

In this μ_i is a fuzzy set, x_i is constant.

Defuzzification

Inference gives the fussy set as output. This fuzzy output must be converted to the crisp set. This method of conversion is called defuzzification. The methods of defuzzification are.

The Max criterion method:

It gives the point where the membership control action arrives at a peak point.

The Height method:

The average of each centroid, weighted by height, is used to calculate the final Lo, which is expressed as follows.

$$L_o = \frac{\sum_{i=1}^n u_i \mu(u_i)}{\sum_{i=1}^n \mu(u_i)} \quad (5)$$

Centroid method or Centre of Area method (COA):

This method is widely used. In this method the centroid of the area

formed by the membership function is given as fuzzy output.

$$Y = \frac{\int \mu_y(y) y dy}{\int \mu_y(y) dy} \quad (6)$$

Fuzzy logic based ITSAC

The RSC and GSC of DFIG is controlled using ITSAC. This ITSAC control block parameters are to be chosen carefully and they are required to be auto tuned according to system status. This auto tuning helps in timely control of system under severe disturbances thereby avoiding the aggregation of system deterioration. For tuning of the ITSAC controller parameters automatically and to ensure self-adaptation, a FLC is used. The parameters of the ITSAC controller η_i , $i = 1$ to 8 ($\beta_{1,k,j}$, $\beta_{2,k,j}$) were adjusted and tuned by $\Delta\eta_i$ as illustrated in Fig.2. The controller parameters are corrected by using the errors of the d-q components of the rotor currents (S_{dq}). The adjusted variables are estimated and tuned as follows:

$$\text{Tuned Variable} = \eta_i + \Delta\eta_i, \quad i = 1, \dots, 8$$

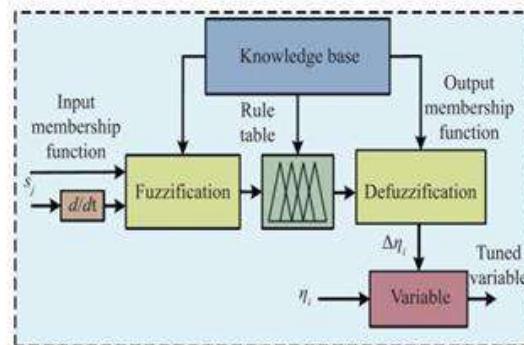


Fig 3: Block diagram of Fuzzy approach

The fuzzy structure used to control ITSAC consists of two inputs and one output. Here the input 1 is s_j , input 2 is Δs_j and output is $\Delta\eta_i$. The fuzzy inference system used here is MAMDANI as shown in figure 3. Here the centroid method of defuzzification is used.

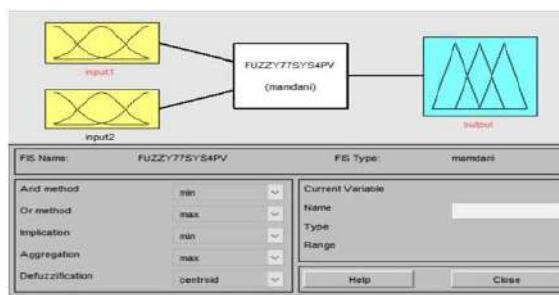


Fig 4: Fuzzy logic structure

The fuzzy logic controller[9-11] inputs s_j , Δs_j and ouput $\Delta \eta_i$ and their membership functions are illustrated in Fig. 5 to 7 respectively.

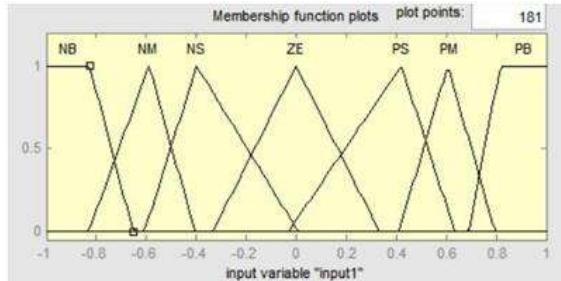


Fig 5. Input 1 s_j

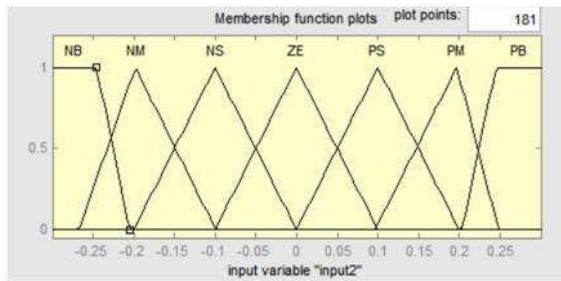


Fig 6. Input 2 Δs_j

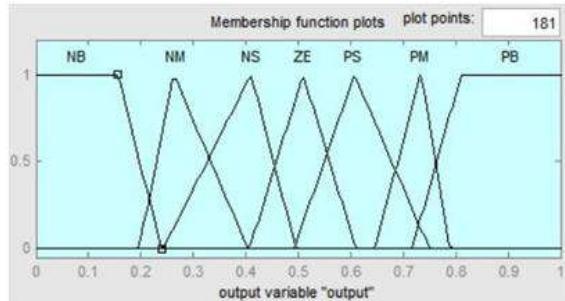


Fig 7. Output $\Delta \eta_i$

This membership function has seven segments. They are large positive (PB), medium positive (PM), small positive (PS), zero (ZE), small negative (NS), medium negative (NM), and large negative (NB) It is shown in table 3.

In this we are using 49 fuzzy rules to implement the ITSAC controller in the DFIG wind turbine. Some of the rules used in the simulation results are shown in the table3.

Table 3. Control rules for $\Delta\eta_i$

		S						
		PB	PM	PS	ZE	NS	NM	NB
Δs	PB	NB	NB	NM	NM	NM	ZE	ZE
	PM	PB	PB	PM	PS	PS	ZE	ZE
	PS	PB	PM	PS	PS	ZE	NS	NM
	ZE	PM	PM	PS	ZE	NS	NM	NM
	NS	PS	PS	ZE	NS	NS	NM	NB
	NM	ZE	ZE	NS	NS	NM	NB	NB
	NB	ZE	ZE	NS	NM	NM	NB	NB

3 Simulation Results

The topography of a DFIG-based wind turbine (WT) connected to the microgrid is described in ITSMD controlled system of DFIG connected to micro grid is represented diagrammatically as shown in Fig.8.The wind turbine parameters considered are tabulated in Table4.

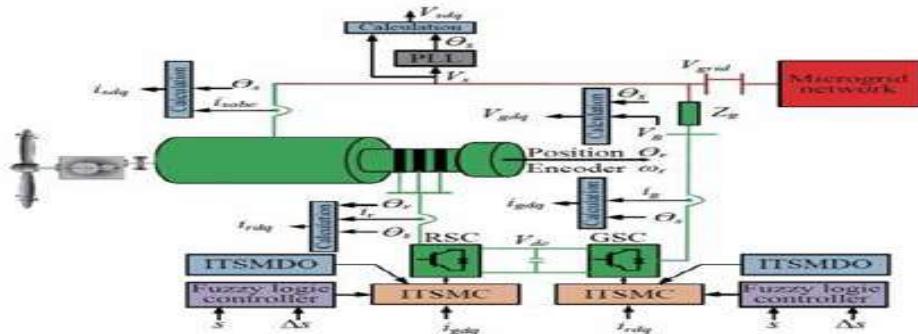


Fig: 8 Performance evaluation of Fuzzy based.

Table 4: Parameters of DFIG.

Parameter	Range
Power	4.5Mw
Voltage	575V

Frequency	60Hz
$R_s / L_s / M$	0.00706(pu)/0.171(pu)/2.6(pu)
L_r / R_r	0.156(pu)/0.005(pu)
L_g / R_g	0.0622(mH)/0.1732(milli ohms)
DC link voltage	1200V
Dc link capacitor	0.03F

Performance evaluation of Fuzzy controller based ITSMC approach

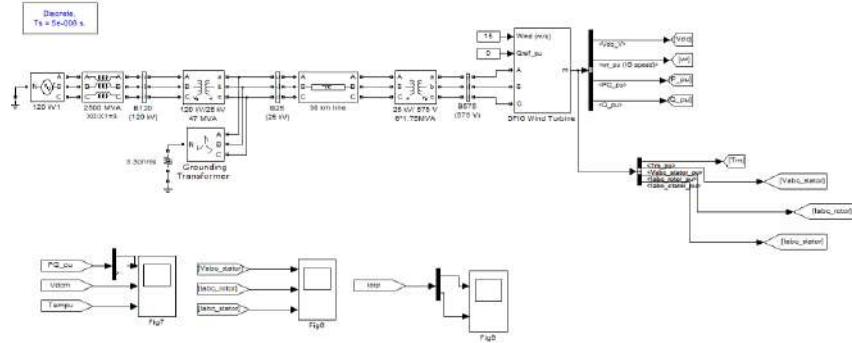


Fig 9: Simulink block diagram of DFIG based WT connected to micro-grid

Performance evaluation of Robustness to parametric variations: In this section testing and comparing the performance of the DFIG under fuzzy based ITSMC, SSMC approaches due to parametric variations was made. The rotor, stator inductance and resistance values of DFIG were allowed to a 20 % increase in their corresponding original values already given in Table 2. The variations in DFIG's voltage of DC- link and real power due to parameter variations are compared. In Figs. 10(a)–(b) comparisons are made among normal parameters, fuzzy based ITSMC due to parametric variation. In Figs.11 (a)–(b) comparisons are made among normal parameters, SSMC due to parametric variation.

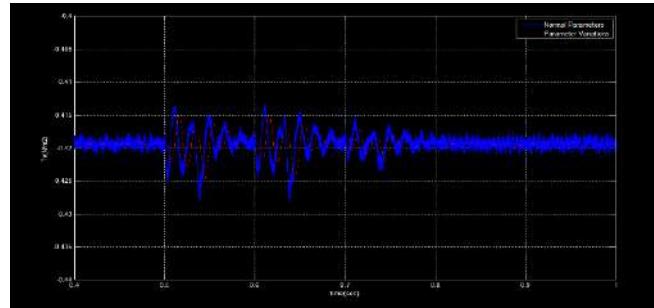


Fig 10(a): Comparison between the normal and parametric variations of Electromagnetic torque using fuzzy based ITSMC

From Fig 10(a) when 20% parametric variations are made in DFIG in wind turbine the variations in electromagnetic torque between normal and when parametric variations occur the limits of variations are within the limit in the fuzzy based ITSMC approach like variations are within small limits. So if small variations are occur in the parameters of DFIG we don't need to disconnect the wind turbine from the grid since the variations are with in the limit.

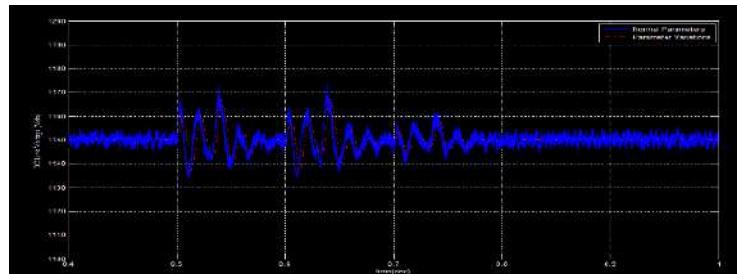


Fig 10(b): Comparison between the normal and parameter variations of DC link voltage while using fuzzy based ITSMC controller

From Fig 10(b) when 20% parametric variations are made in DFIG in wind turbine the variations in DC link voltage which is the output this variations between normal and when parametric variations occur the limits of variations are within the limit in the fuzzy based ITSMC approach like variations are within small limits. So if small variations are occur in the parameters of DFIG we don't need to remove the wind turbine from the grid since the variations are with-in the limit.

When we done the parameter variations in the SSMC controller by 20% variations in the DFIG wind turbine parameters the variations of electro magnetic torque and the dc link voltages are not in acceptable limit so that the converters which are

present near to the wind turbine may get damage. So we need to disconnect the wind turbine from the grid it may causes the interruption of power generation in the grid and also causes the failure of DFIG.

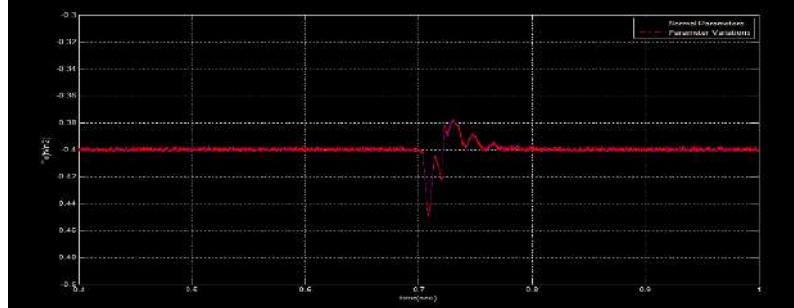


Fig 11(a): Comparison between the normal and parameter variations of Electro Magnetic Torque while using SSMC controller.

In fig 11(a) the variations are occurred in the electromagnetic Torque due to 20% parametric variations in the DFIG when we compare this two cases in SSMC the variations between the normal and SSMC controller in parametric variations the limits of electromagnetic Torque are not within the acceptable limits so it is not suitable to continuously connecting the DFIG to the grid, so we need to disconnect from the grid while we are using the SSMC controller.

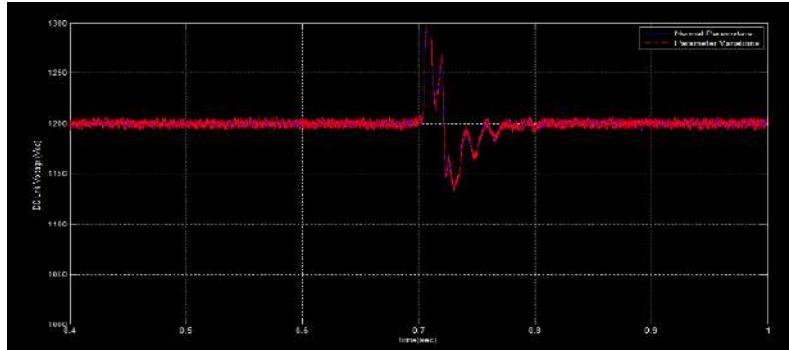


Fig 11(b): Comparison between the Normal and Parameter variations of DC link voltage while using SSMC controller.

Similarly In fig 11(b) the variations are occurred in the DC link voltage due to 20% parametric variations in the DFIG when we compare this two cases in SSMC the variations between the normal and SSMC controller in parametric variations the limits of DC link voltage are not within the acceptable limits so it is not suitable to continuously connecting the DFIG to the grid, so we need to disconnect from the grid while we are using the SSMC controller.

4. CONCLUSION.

To improve the power quality of wind turbines under unbalance voltage conditions, an integrated final sliding mode controller design is presented. The design combines the interferometers' evaluation capabilities with the integrated sliding-mode controller of the terminal's robustness, quick response, and short-term high-resolution properties. Both RSC and GSC have successfully used DFIG-based wind turbines. A fuzzy logic technique is used to automatically set the controller gain. Performance was assessed in a variety of settings, including those with a significant voltage drop. Its dynamic response was contrasted with an ordinary SMC's. Results from performance analysis and simulation demonstrated that it can maintain electromagnetic pairs, active power, currents, and DC link voltages within acceptable bounds even under the most extreme unbalanced voltage conditions.

References

1. S T. Ackermann and P. E. Morthorst, "Economic aspects of wind power in power systems," Wind Power in Power Systems, pp. 383, Aug.2005.
2. M. Tsili and S. Papathanassiou, "A review of grid code technical requirements for wind farms," IET Renewable Power Generation, vol. 3, no. 3, pp. 308;332, Sep. 2009.
3. J. Yao, H. Li, Z. Chen, X. Xia, X. Chen, Q. Li, and Y. Liao, "Enhanced control of a DFIG-based wind-power generation system with series grid-side converter under unbalanced grid voltage conditions," IEEE Transactions on Power Electronics, vol. 28, no. 7, pp. 3167;3181, July 2013.
4. B Hariprasad, Dr.P.Bharat Kumar, Dr.P.Sujatha and Dr.G.Sreenivasan, " A Novel Adaptive Rule-Based Islanding Detection Technique for Voltage Source Inverter-Based Distribution Generation ", in Journal of Green Engineering (JGE), Online ISSN: 1904-4720 .vol.11, pp.511-529, March 2021.
5. S. Seman, J. Niiranen, and A. Arkkio, "Ride-through analysis of doubly fed induction wind-power generator under unsymmetrical network disturbance," IEEE Transactions on Power Systems, vol. 21, no. 4, pp. 1782;1789, Nov. 2006.
6. B Hariprasad, Dr.P.Bharat Kumar, Dr.P.Sujatha and Dr.G.Sreenivasan, "Island Detection In Inverter Based Distributed Generation Using A Hybrid Method", in Second International Conference on Artificial Intelligence and Smart Energy (ICAIS-2022) IEEE Xplore Part Number: CFP22OAB-ART; ISBN: 978-1-6654-0052-7, doi: 10.1109/ICAIS53314.2022.9742815.
7. S. Zhang, K. Tseng, S. S. Choi, T. D. Nguyen, and D. L. Yao, "Advanced control of series voltage compensation to enhance wind turbine ride through," EEE Transactions on Power Electronics, vol. 27, no. 2, pp. 763;772, Feb 2012.
8. S. W. Mohod and M. V. Aware, "A STATCOM-control scheme for grid connected wind energy system for power quality improvement," IEEE Systems Journal, vol. 4, no. 3, pp.

346;352, Sep. 2010.

9. B Hariprasad, Dr.P.Bharat Kumar, Dr.P.Sujatha and Dr.G.Sreenivasan, " Review Of Islanding Detection Methods For Distribution Generation", in International Conference on Breakthrough in Heuristics and Reciprocalation of Advanced Technologies, BHARAT 2022 IEEE Xplore ISBN: 978-1-6654-3626-7. doi:10.1109/BHARAT53139.2022.00029.
10. M. Edrah, K. L. Lo, and O. Anaya-Lara, "Impacts of high penetration of DFIG wind turbines on rotor angle stability of power systems," IEEE Transactions on Sustainable Energy, vol. 6, no. 3, pp. 759;766, July 2015.
11. B Hariprasad, Dr.P.Bharat Kumar, Dr.P.Sujatha and Dr.G.Sreenivasan, " A Novel Hybrid (Active & Passive) Islanding Detection Method for Distributed Generation" in International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies, vol.13, issue.9, pp.1-16, June 2022. doi: 10.14456/ITJEMAST.2022.174.

NAML- A Novel Approach of Machine Learning Implementation in the Hospitality Industry

Ashwin C S, Dr. Sheela Thavasi, Rangarajan KR
Sri Sai Ram Engineering College, Information Technology,
USA, INDIA
tocsashwin@gmail.com, hod.it@sairam.edu.in, ranga1290@gmail.com

Abstract— Machine Learning has been utilized for the improvement of consumer experience, operational analytics, and revenue management in the hospitality industry. The paper discusses the use of ML in engaging the customers in the industry. ML also improves the ways people make decisions in the hospitality industry.

Keywords—Machine learning, hospitality, algorithms, automation

I. INTRODUCTION

The hospitality industry is moving continuously. Not many industries are as customer centric as the hospitality industry, with all types of businesses trying to develop a valuable journey for customers from beginning to end. The utilization of Machine Learning has the capability of transforming the hospitality industry. Harnessing the power of Machine Learning via Data Science platforms assists in creating a more personalized view of all customers that support associating them on a deep level than before. Machine Learning technology aims at assembling the arrangement of collecting data and learning from the data and improves self-capacity via experiences without the engagement of humans and reprogramming [1].

Machine Learning has been utilized for the improvement of consumer experience, operational analytics, and revenue management in the hospitality industry. Data aids organizations in fulfilling the increasing expectations of the customers when presented with new, unexpected problems in the hospitality industry. Capturing data regarding consumer purchases, consumer experience inquiries, travel patterns, and location preferences also assist in increasing the efficiency of the analytics being utilized. Engaging with the customers and informing them regarding the various offers and felicitating them on any special day can be an approach for involving consumers [2].

Machine Learning improves consumer services. So, hotels have installed Machine learning for smoothly doing this job. Today, robots take place in that position to assist humans in completing any task as quickly as possible, including concierge services, room services, and housekeeping are performed by robots. Modern technology has modified human employment. Hotels nowadays provide self-service

approaches to consumers. In this article, the use of ML in the hospitality industry and its innovative approaches in data analysis and modeling are discussed. The improvement in the industry with machine learning is also discussed with proper examples and justification. It is also proved that machine learning improves the revenue of the hospitality industry [2].

II. RELATED WORK

USEFULNESSES OF MACHINE LEARNING IN THE HOSPITALITY INDUSTRY

Users generate a vast amount of data during the search for destinations, hotels, and flights for travel. These are metadata (browser, device, location, referral source, and duration), Behavioral data (Search enquiries, travel booking history, email subscription, destination searches, and other online activities), CRM data (Service preference, purchasing details), Social media data (Ratings, service feedbacks, geo-tagged locations, shared photos, and social media comments). Machine learning is very efficient in gathering and processing this data to help a company and the hospitality industry in different ways [2].

Automated processes and services: It is known to everyone that it is so frustrating when half an hour is taken to reach customer service to reschedule a booking. Generally, customers change their choice while facing this type of issue. By using AI-based assistants, that is, chatbots, the hospitality industry can reduce the average time of response to just a few minutes [3].

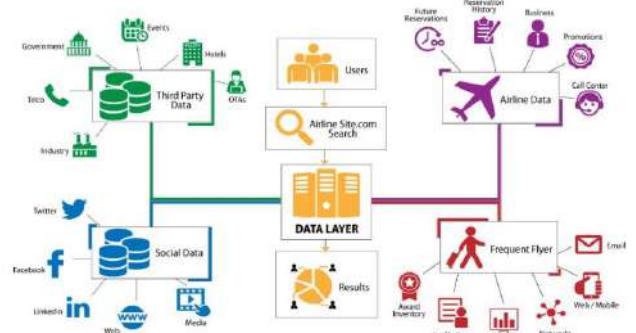


Figure 1. Automated system in hospitality industry

The data layer is composed of third-party, social data, airline data, and frequent flyer data. Data shows that most hospitality businesses are eager to implement this type of self-service technology to manage customer support and repetitive tasks. AI-based solutions are also helpful for resource planning and revenue accounting using data layer [3].

Personalization of customer experiences: Lots of random travel offers cannot be sent to a user with an email which is also embarrassing. Implementation of AI in the hospitality industry provides insight into customer behavior which gives the company the opportunity to tailor-made deals according to the family status of the customer, favorite cities, the purpose of visit, and hotel location preference. The chances of catching customers' attention increase multiply. On the other hand, people love to express their views online.

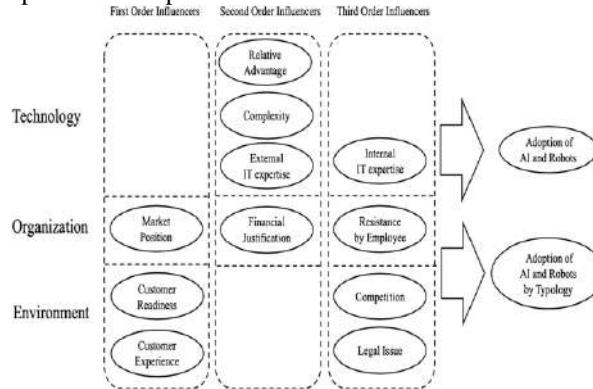


Figure 2. Adoption of ML in hospitality industry

The technology, organization, and Environment of the ML system is described in the diagram. AI-powered software analyses guest preferences with their tracking patterns of them, and staffs get real-time recommendations from the software. This results in the improvement of customer satisfaction and building loyalty [4].

Improvement of staff efficiency: Customers love the service when the staff of the hotel remembers them and their preferences. The whole customer management system of a company gets a personal touch when a company utilizes the capacity of machine learning in it. Natural language processing and Speech analytics apps can be used to extract meaningful data from customers and support staff voice interactions. Employees can also improve their work efficiency in their free time, generated using machine learning tools [4].

Detecting fraudulent activities: Many hotels face the issue of being scammed by customers who demand a chargeback from the bank as they claim one of their cards was stolen or their other credentials were stolen. Billions of monetary funds which are invested in dollars, or any other currency can be saved by the hospitality industry by using artificial intelligence to find out the illegal activity by evaluating transactions [5].

Optimization of expenses: Everyday technical malfunctions and natural disasters cause lots of delays in the hospitality industry. Thanks to the advanced algorithm of machine learning, predictions can possibly be made about travel disruptions based on flight delay and weather

information. This helps to reduce losses for carriers and travelers. Many hospitality management companies use machine learning and artificial intelligence combined with trackers to plan regular activities [5].

Creating more value: A suitable algorithm can help to provide better service with an increase in profit by analyzing the customer activities at the hotel, such as room preference, room service choices etc. Analysis of travel goals, value and price preference, destination, and time spent on the hotel by algorithms gives the whole hospitality management system more value with better customer satisfaction, timesaving, and cost reduction. The management's insight into customers' demand and choices improve. Thus, the management can invest time and effort in other business sections [7].

III. PROPOSED WORK

HIGH-TECH INNOVATION IN HOSPITALITY INDUSTRY USING MACHINE LEARNING

In the hospitality industry, there is a demand of innovative technology that improves success in handling operations. The fundamental requirement of the industry is to analyze customer behavior that requires huge statistics. With the help of machine learning algorithms and techniques, it is possible to access huge amount of data associated with the hospitality business. ML helps in collecting and analyze tourist's behavior by evaluating their data. This includes monitoring their stay, relevancy of the service in the hospitality industry and so on. In the era of internet collection and monitoring of such data is very easy. However, with the intervention of machine learning algorithms, the evaluation of the data using

ML models can be possible. The models create networking between the user and the industry as well which is a great form of innovation. Currently, the industry is looking forward innovation through mobile apps, online user reviews, video investigations, etc. After using a service, people generally put review in form of ratings and comments [7]. This user data is collected automatically by the machine learning algorithms and run it with the help of effective models to make decisions in the industry. The revelations of the innovation started explicitly in 1970s when the computer reservation system has started. With the intervention of global distribution system, it became possible to develop innovation in the hospitality industry [6].

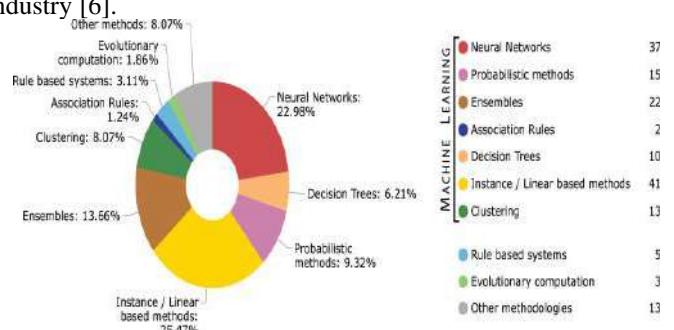


Figure 3. systematic evaluation of ML components in hospitality industry

The machine learning types are classified in this diagram. Sentimental Analysis Technologies and Natural Language Processing can analyze this vast amount of user reviews from social media and websites. An efficient software can also identify these posts' emotional backgrounds. Thus, a company can notice both the negative and positive sides of its business and improve its service accordingly [3]. Machine learning is associated with the innovative data science technique that introduces maximum tangible service to the hospitality industry. It helps in categorizing the input objects and estimate the output as a statistical model. The use of training and testing of data through models and algorithms help in dividing data and analyzing trends. For example, analyzing sales and customer behavior becomes very easy. Usually, the machine learning algorithm is applied in four ordinary forms, namely supervised, unsupervised, semi-supervised, and reinforcement learning. In this way, focus is given to the customers engagement and loyalty. Machine Learning helps the consumers in the industry to engage customers and loyalty for the businesses. This improves the overall industry's performance. So, it can be said that machine learning brings high-tech innovation to the industry [6].

ML is convenient in every way. Even new users can use it as it is user-friendly and environment friendly. It is a convenient way of providing accurate information by sourcing data from several companies. The tourist's reviews and comments are evaluated suing modern models that are coming into the industry every day. With new interventions, the machine learning algorithm brings new techniques of evaluating data and make the system completely automatic. Moreover, machine learning brings automation which is a high-tech innovation in today's world [9].

By evaluating the historic data in the industry, it is important to innovation automation. ML helps in high level of decision making. ML improves customer service, customer experience, and relieve the industry from extra pressure and wrong decision making. The system helps in innovating the booking and occupancy system, improves segmentation and demand of the customers. With high-tech innovation, ML can improve the management and operations. It also helps in brand monitoring and uses advanced competition analysis in the industry. The employees are associated with technological support as well that improves their pricing and performance statistics. ML in the tourism industry is most valuable for a company when it analyses the competition in the market [8].

ML is based on the hyperparameters as the machine starts learning in a better way. In the hospitality industry, there is a need of mechanized system that is advantageous for both the customers and the service providers. The design of ML algorithm is done to solve different actions in the hospitality industry. It helps in future predictions of the company's trends. data is analyzed with structured and semi-structured pattern for future predictions. Artificial Intelligence helps in designing and development of algorithms as well. High-tech innovations like clustering and classification algorithms help the hospitality industry to grow. ML also improves the pre-processing of data [7].

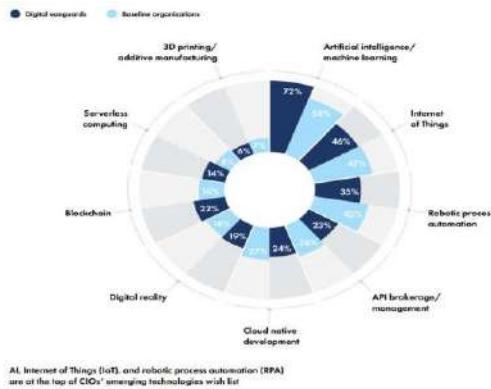


Figure 4. State of AI in hospitality industry

The diagram describes the state of AI in the hospitality industry.

Machine Learning facilitates check-out and check-in with facial recognition. Facial recognition permits them to check out and check in easily without any waste of time. So, there is no requirement for waiting at the front desk. Machine Learning techniques permit employees in being part of automated management systems, where managers and employees predict the future of any business condition via consumer reviews or collected data and plan the preparation for the future operations [7]. Machine Learning can support associating hotel management and consumers from one viewpoint, where both understand and recognize the existing reactions and smart services. It allows them in finding out undiscovered information from collected data. So, hotels involve Machine Learning systems for accurately doing tasks, including occupancy, and booking, segmentation, revenue management, yields management, demand forecasting, pricing, competitive analysis, brand monitoring, performance evaluation, and data collection from external and internal sources for understanding consumer behaviors. Nowadays, Machine Learning has been at the state-of-the-art of practices of revenue management in the hospitality industry. The basic concept of Machine Learning is that the computer learns from data and develops a model automatically by normalizing the pattern identified from the data. Machine Learning algorithms develop a model dependent on sample data, referred to as training data, for making decisions or predictions without being programmed for doing so. Machine Learning is computer algorithms that learn automatically from data to discover insights or hidden patterns [8].

IV. RESULT ANALYSIS

ML IN REVENUE MANAGEMENT IN THE HOSPITALITY INDUSTRY

For an organization in the hospitality industry, it is important to improve the revenue collection by shaping the demand in the market. The complex activity of revenue management deals in customer segmentation, demand forecasting, pricing techniques and yield management. With such techniques, the application of ML becomes highly innovative and dynamic. There are several components of ML

that improves the revenue of the company, such as analyzing customer behavior and demographics. With the help of data collection of the customers, the company can improve demand in the market for the hotels [11].



Figure 5. ML revenue management in hospitality industry

Advanced revenue management systems can utilize Machine Learning models in different areas of revenue management, such as pricing, demand forecasting, customized guest services, and market segmentation, with recent development in Machine Learning technologies supported by big data [12]. Machine Learning methods are broadly utilized in several application areas, including speech recognition and image processing, whereas conventional algorithms cannot be capable of solving the larger-scale, often complex issues. Revenue managers do not require to put any number into a spreadsheet with machine-recommended pricing and data-driven forecasts of consumer demands. Revenue management systems can greatly be automated by leveraging advancement in Machine Learning technology. Machine Learning procedures manage the operations, and the organizations investigate new data regarding consumer demands, behaviors, and future trends in the hospitality industry [9].

In the hotel industry, dynamic pricing is applied by advanced machine learning strategies. ML uses different technologies to innovate the use of models for revenue analysis. It helps in analyzing the competition by using trends analysis. The revenue management principles remain the same regardless of the underlying software used in the standard industry. A tangible quality shift can be seen in this case where science-based techniques are used in the revenue management system. Machine learning also entails the training and testing of statistical models. It helps in forecasting the values in a supervised as well as unsupervised ways [9].

ML uses new generation solutions like maintaining accuracy, autonomy, and integrity. The real-time data processing system helps the hospitality company to redefine revenue management. It also enables customer segmentation in an automated way. Machine Learning helps in modernizing underlying techniques in a tangible way that improves the efficiency of the company. ML improves revenue by demand forecasting [10].

V. CONCLUSION

The use of Machine Learning (ML) algorithm is increasing day by day in the hospitality industry because of its effectiveness in automating the system. ML is also very informative to the hospitality industry. It is learned from the article that Machine Learning supports associating hotel management and consumers from one viewpoint, where both

understand and recognize the existing reactions and smart services. ML provides top benefits to the hospitality industry by revenue analysis, building models for detection of trends, forecasting data, etc.

REFERENCES

- [1] Alotaibi, Eid. "Application of machine learning in the hotel industry: a critical review." *Journal of Association of Arab Universities for Tourism and Hospitality* 18, no. 3 (2020): 78-96.
- [2] Aluri, Ajay, Bradley S. Price, and Nancy H. McIntyre. "Using machine learning to co-create value through dynamic customer engagement in a brand loyalty program." *Journal of Hospitality & Tourism Research* 43, no. 1 (2019): 78-100.
- [3] Antonio, Nuno, Ana de Almeida, and Luis Nunes. "Predicting hotel bookings cancellation with a machine learning classification model." In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1049-1054. IEEE, 2017.
- [4] Caicedo-Torres, William, and Fabián Payares. "A machine learning model for occupancy rates and demand forecasting in the hospitality industry." In *Ibero-American Conference on Artificial Intelligence*, pp. 201-211. Springer, Cham, 2016.
- [5] Go, Hanyoung, Myunghwa Kang, and SeungBeum Chris Suh. "Machine learning of robots in tourism and hospitality: interactive technology acceptance model (iTAM)—cutting edge." *Tourism Review* (2020).
- [6] Mahesh, Batta. "Machine learning algorithms-a review." *International Journal of Science and Research (IJSR) [Internet]* 9 (2020): 381-386.
- [7] Mbunge, Elliot, and Benhildah Muchemwa. "Deep Learning and Machine Learning Techniques for Analyzing Travelers' Online Reviews: A Review." *Optimizing Digital Solutions for Hyper-Personalization in Tourism and Hospitality* (2022): 20-39.
- [8] Park, Eunhye Olivia, Bongsug Chae, and Junehee Kwon. "Toward understanding the topical structure of hospitality literature: Applying machine learning and traditional statistics." *International Journal of Contemporary Hospitality Management* (2018).
- [9] Parvez, M. Omar. "Use of machine learning technology for tourist and organizational services: high-tech innovation in the hospitality industry." *Journal of Tourism Futures* (2020).
- [10] Satu, Md Shahriare, Khair Ahammed, and Mohammad Zoynul Abedin. "Performance Analysis of Machine Learning Techniques to Predict Hotel booking Cancellations in Hospitality Industry." In *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pp. 1-6. IEEE, 2020.
- [11] Vargas-Calderón, Vladimir, Andreina Moros Ochoa, Gilmer Yovani Castro Nieto, and Jorge E. Camargo. "Machine learning for assessing quality of service in the hospitality sector based on customer reviews." *Information Technology & Tourism* 23, no. 3 (2021): 351-379.

- [11] Zhou, Zhi-Hua. *Machine learning*. Springer Nature, 2021.
- [12] Qu, Kaiyang, Fei Guo, Xiangrong Liu, Yuan Lin, and Quan Zou. "Application of machine learning in microbiology." *Frontiers in microbiology* 10 (2019): 827.

CISUM: Novel research on Cloud Computing Simulators and Future Scope for Computational Research

Ashwin C S, VKG Kalaiselvi, Rangarajan KR
Sri Sai Ram Engineering College, Information Technology,
USA , INDIA
tocsashwin@gmail.com, Kalaiselvi.IT@sairam.edu.in, ranga1290@gmail.com

Abstract- For the primary purpose of this research, key emphasis is on comprehensively reviewing existing literature on the usage of cloud computing simulators. For concluding this research, several types of cloud simulators are identified, reviewed and compared with one another based on specific criteria of technical features. As concluded in the research, the most crucial aspects related to cloud simulation is the scope of accessing it from any location and hence, presenting enhanced flexibility for the users, delocalising staff training, and ensuring learning on basis of collaborative and dynamic learning. Hence, the future scope of simulation lies in the system of cloud computing with a possibility that all engineering simulations will be conducted using cloud computing.

Keywords- *Cloud computing, cloud computing simulation, simulators, and computation*

I. INTRODUCTION

Cloud computing broadly refers to the services and applications that run on a distributed network using internet protocols and virtualised resources, which are key requisites for it to operate [1]. Since recent times, cloud computing has rapidly developed while gaining the attention of several researchers, practitioners, and investors. This can be regarded as an outcome of the evolution of distributed client-server computing from centralised mainframe computers [1]. However, even though the commencement of cloud computing has taken place since the past decade, it is still in its early stages, considering the challenges encountered in it. Moreover, setting up a physical cloud for experiments and research on cloud computing may not be feasible and economical for industries and research institutions [2]. Therefore, simulations have emerged as a common tool to infer the knowledge of impact design choices in systems, as straight-forward experiments may not be realistic and extremely expensive. Hence, simulators play an important role in reducing costs, efforts, and chances of malfunction, while boosting confidence [2]. The usage of computing simulators assists in designing and foreseeing the performance of systems, structures and machines, without actually building components in the real world.

Cloud simulators help to model different types of cloud applications by creating virtual machines, data centres, as well as other utilities that can be properly constructed. There are several cloud simulators that have been proposed and developed by different institutions for conducting cloud research [3]. These simulators have variety of features that differentiate them from each other, such as user interface,

base programming language, licensing, and extensibility [3]. The aim of this research is to comprehensively review

existing literature on the use of cloud computing simulators, thereby focusing on their features, and analysing their strengths and drawbacks. Several simulators of cloud computing are compared on the basis of specific criteria.

II. RELATED WORKS

Since the last few years, cloud computing has been gaining significant popularity as it presents an efficient and flexible solution for several services by using the internet [4]. Cloud is an extremely complex and large system as it comprises of various storage technologies, internet latencies, bandwidths, algorithms of task scheduling, service brokers, physical machines, service providers and several users. On the contrary, there is a need for several configurations in order to succeed in cloud based implementations [4]. Therefore, it is extremely challenging for researchers to successfully assess policy performances in the actual cloud environment. The tool of cloud simulation is an attractive and viable solutions for the challenge as it provides scope to analyse system behaviour and consider components in several scenarios [5].

The environments of cloud computing have certain attributes such as scalability, dynamicity, and heterogeneity which hold the requirement of specific resource tools. The real performance of cloud experiments are challenges by certain limitations related to hardware and software rescaling and reconfiguration. These end up impeding the performance delivered by the entire system [6]. Service researchers and providers should be tuning their proposed method of cloud computing in various scenarios with several resources for realising its full potential prior to its use across the real environment. Simulation is a technique and science used for making a process model or system, designed specifically for the purpose of testing and evaluating the strategies. Simulation focuses on studying and understanding system behaviour or considering the evaluation of several strategies [6]. Simulators play a crucial role to design the model of the entire system and conduct model experiments. The model can be used further ahead in order to understand system behaviour or evaluate several strategies for system operability.

Traditional system simulators cannot model the community of cloud computing and hence, several researchers focus on

this concern while designing toolkits of simulation for cloud computing [7]. Nevertheless, as several cloud simulators are available, it is crucial to evaluate these tools for selecting the most suitable ones for future systems in computational research [7]. Several cloud tools have been surveyed in the past and majority of the researchers investigated the high level attributes and architecture of all of these simulators.

A. Benefits of Cloud Simulation

The utilisation of cloud simulator assists to model various types of cloud application by creating virtual machines, data centres, and additional utilities for proper construction. Therefore, it makes a stress free environment for analysis. Till the current scenario, several cloud simulators are developed, proposed and used by several institutions for computing research [8]. These developed and proposed simulators have various features which differentiate all of them from one another. Some of these features are extensibility, base language of programming, licensing, and user interface. There are several researchers in existing literature investigating different types of cloud simulators. The utilisation of simulator comes with a number of benefits like minimised cost, in such a manner that rather than purchasing costly software and hardware, there is availability of various simulators free of cost [8]. In addition, the utilisation of simulator helps in easy controllable and repeatable experiments. Since there is simulation of experiments, there is no specific requirement to set up the entire physical system. Simulators also create a friendly environment for easily evaluating performance in several scenarios involved various conditions.

B. Challenges in Cloud Simulation

Majority of the challenges faced during the development of cloud computing by software architects are also faced by cloud simulators. Among these challenges is to determine approaches for routing requests of networking and rerouting for the reduction of delayed arrival [9]. Therefore, cloud simulators have to emphasise upon proposing alternates in order to solve traffic concerns such as the ones existing in data centres of real cloud. Utilisation of relay switches, management of traffic with central control, routing on the

IV. RESULTS ANALYSIS

A. Comparison of Cloud Simulators

S.No.	Cloud Computing Simulators	Underlying Platform	Software/ Hardware Companionship	Programming Language	Types of Services (SaaS, PaaS and IaaS)
1.	CloudSim	SimJava	Software	Java	IaaS
2.	CloudAnalyst	CloudSim	Software	Java	IaaS
3.	GreenCloud	NS2	Both	C++ / oTcl	IaaS
4.	iCanCloud	SIMCAN,	Software	C++	IaaS

basis of service, or enabling decision making for user can include such alternates [10]. Other challenges involved in the network model of cloud simulation are significant to the selection of flexible and fixed mechanism of bandwidth allocation. In addition, the simulation design should be considering the topology of interconnection. In the real environment of cloud computing, a well connected model is unusual and hence, simulators should be modelling realistic networks by the addition of access level, aggregate and additional root level switches. There can also be several internal and external challenges in designing the method of resource allocation for the cloud environment [11]. External challenges include dynamic users in demand of resources, geographical challenges, and optimisation of cost model for maximising revenue. Hence, such challenges can result in limiting the location of virtual machine.

III. PROPOSED WORK

For the purpose of presenting this research, key emphasis is laid upon comprehensively reviewing existing literature on the usage of cloud computing simulators. For concluding this research, several types of cloud simulators will be identified, reviewed and compared with one another based on specific criteria of technical features. An online desk based research is conducted for reviewing existing literature related to different types of cloud simulators. Secondary data is the data collected by others and using them in context with a specific research topic.

Hence, a qualitative methodology has been adopted in this research for achieving the research aim and presenting relevant discussion regarding the same. In qualitative methodology, descriptive texts and data are used rather than using numerical data and statistical analysis for collecting and analysing data. Credible sources will be used in the research for ensuring reliable and valid information is collected in accordance with usage of cloud computing simulators. The comparison of data collected on different cloud simulators will be done on the basis of four factors; underlying platform, companionship of hardware or software, programming language, and the types of services (SaaS, PaaS, and IaaS).

		OMNET, MPI			
5.	MDCSim	CSIM	Software	C++ / Java	-
6.	NetworkCloudSim	CloudSim	Software	Java	IaaS
7.	EMUSIM	CloudSim & AEF	Both	Java	IaaS
8.	CloudReport	CloudSim	Software	Java	IaaS and SaaS
9.	CloudSched	-	Software	Java	IaaS
10.	CloudExp	CloudSim	Software	Java	IaaS, SaaS and PaaS
11.	DCSim	-	Software	Java	IaaS and PaaS
12.	ICARO	-	Software	Java	IaaS and PaaS
13.	SPECI	SimKit	Software	Java	IaaS
14.	GroudSim	-	Software	Java	IaaS
15.	SmartSim	CloudSim	Software	Java	IaaS
16.	SimIC	SimJava	Software	Java	-
17.	DynamicCloudSim	CloudSim	Software	Java	IaaS
18.	CloudSimSDN	CloudSim	Software	Java	IaaS
19.	SecCloudSim	iCanCloud	Software	C++	IaaS
20.	CEPSim	CloudSim	Software	Java	IaaS
21.	PICS	-	Software	Python	IaaS
22.	TeachCloud	CloudSim	Software	Java	IaaS
23.	CDOSim	CloudSim	Software	Java	IaaS
24.	CloudNetSim++	CloudSim	Software	Java	IaaS
25.	DartCSim+	CloudSim	Software	Java	IaaS

26.	GDCSim	Bluetool	Software	C++/XML	IaaS
27.	FlexCloud	-	Software	Java	IaaS
28.	VirtualCloud	-	Software	XML	IaaS
29.	Cloud2Sim	CloudSim	Software	Java	IaaS
30.	DesktopCloudSim	CloudSim	Software	Java	IaaS
31.	WorkflowSim	CloudSim	Software	Java	IaaS
32.	CloudMIG	CloudSim	Software	Java	IaaS and PaaS
33.	EduCloud	-	Both	Java	IaaS

B. Comparative Discussion

All of the compared simulators are clearly supporting various quality metrics. All the simulators support a combination of quality metrics which are different from other simulators but share the scope to implement extension of quality metrics with additional metrics on the basis of required problems to be address [12]. The simulators also suggest optimum configuration in the elements of cloud for supporting the environment under investigation. All of the investigated simulators relate dynamic operability of experiments that provide the ability to users for changing the configured elements while conducting the experiments [12]. There is also strong support of user preferences for the parameters of experiments so that users can keep them for the future operability of the experiments.

users so that average experiment results can be obtained [13]. 52 per cent of the cloud simulators are developed on the simulator of CloudSim (see Figure 1 for CloudSim architecture). This indicates why Java is among the most common language of programming utilised for the development of cloud simulators. The second base language of programming is C++. Only one simulator, PICS was developed by the use of Python. EduCloud, EMUSIM and GreenCloud are among the sole simulators that involve the combination of hardware modelling and simulation by simulating the cloud software [13].

95 per cent of simulators support the simulation of IaaS, while some of them are supported by the simulation of SaaS and PaaS. Approximately 76 per cent of the cloud simulators under investigation are present under the licensing of download through open source code. Approximately 80 per cent of the cloud simulators compared support modelling on the basis of cost [14]. This is crucial for cloud providers for the examination of new pricing strategies and plans. Some of the simulators do not support the placement of communication for interacting with the cloud components. Majority of the simulators are fast and clearly express results of experiments at an extremely fast pace [14]. 60 per cent of the simulators support modelling of energy through modelling devices, servers and networks. Approximately 70 per cent of the investigated simulators support the modes of power saving or event to collect information related to the power consumer of devices. Approximately 40 per cent of the cloud simulators support the simulator of federation policy as the requirement of studying by the connection of multiple clouds [15].

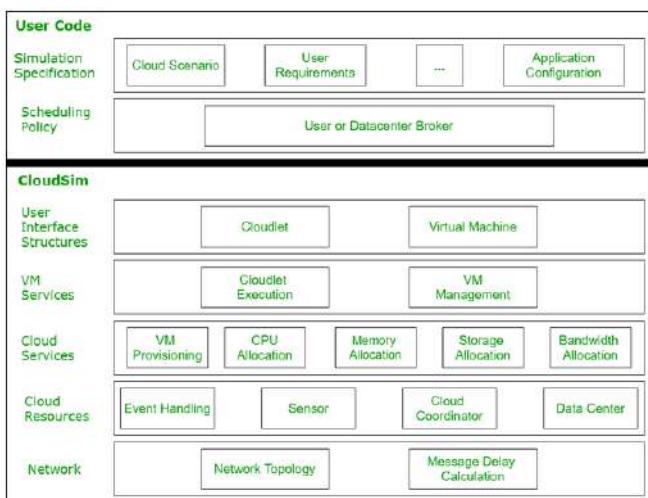


Figure 1: CloudSim Architecture

Only 50 per cent of the cloud simulators have graphical interface of user. Researchers show preference towards simulators for supporting the graphical interface so that there is easy configuration and operability of experiments. All of the compared simulators support operability of experiments for long durations with the save preferences of

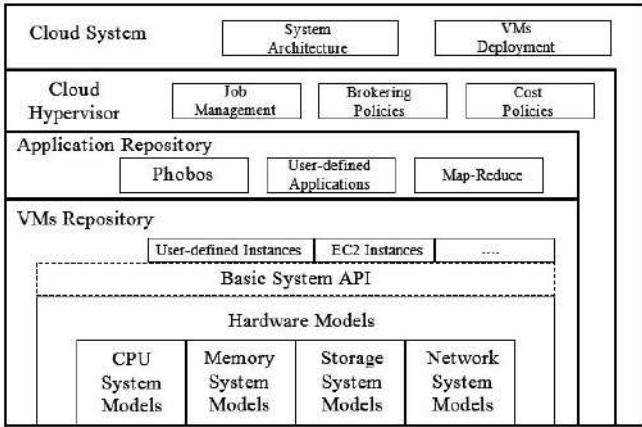


Figure 2: iCanCloud Architecture

Out of all of the simulators, iCanCloud is the sole simulator which plans for obtaining features to operate various independent experiments parallel to the utilisation of resources available for the simulator. All of the compared simulators are known for supporting applicable model by the parameterisation of data transfer, computation and deadlines for support execution [16]. All of the simulators related to network simulation are related to the simulation of packets communicated across the elements of cloud. SecCloudSim and CloudExp are the key simulators that support features of security while modelling communication between cloud components.

C. Future Scope of Cloud Simulation

Cloud computing is a crucial technology that consolidates great levers for the promotion of innovation and consolidation of digitalisation across businesses [16]. The most crucial aspects related to cloud simulation is the scope of accessing it from any location and hence, presenting enhanced flexibility for the users, delocalising staff training, and ensuring learning on basis of collaborative and dynamic learning. In a way, it facilitates co-existence with uncertain situations caused as a result of the pandemic, where it is impossible for making necessary predictions [18]. In addition, by the integration of teams from various locations, cloud simulators are tools used for encouraging team development and promoting group learning.

The utilisation of cloud simulator will further signify reduced costs of personnel mobilisation and travels [17]. It affects several costs as the same simulator is utilised by several business in a delocalised manner, without the transportation of physical simulator to every site. Thus, operations and applications are optimised without the sacrifice of performance in comparison with a simulator of full scope [19]. Hence, the future scope of simulation lies in the system of cloud computing with a possibility that all engineering simulations will be conducted using cloud computing due to its benefits and potential.

V. CONCLUSION

The utilisation of cloud simulator assists to model various types of cloud application by creating virtual machines, data centres, and additional utilities for proper construction. Therefore, it makes a stress free environment for analysis.

Till the current scenario, several cloud simulators are developed, proposed and used by several institutions for computing research. These developed and proposed simulators have various features which differentiate all of them from one another. As per the data analysed in this research, it can be concluded that all of the compared simulators are known for supporting applicable model by the parameterisation of data transfer, computation and deadlines for support execution. All of the simulators related to network simulation are related to the simulation of packets communicated across the elements of cloud. SecCloudSim and CloudExp are the key simulators that support features of security while modelling communication between cloud components.

REFERENCES

- [1] N. Mansouri, Ghafari, R. Zade B.M. “Cloud computing simulators: A comprehensive review,” *Simulation Modelling Practice and Theory*. 104 (2020) 102-144.
- [2] W. Lin, Xu S, He L, Li J. “Multi-resource scheduling and power simulation for cloud computing,” *Information Sciences*. 2017 Aug 1; 397:168-86.
- [3] M.C. Silva Filho, Oliveira RL, Monteiro CC, Inácio PR, Freire MM. “CloudSim Plus: A cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness.” In *2017 IFIP/IEEE symposium on integrated network and service management (IM)* 2017 May 8 (pp. 400-406). IEEE.
- [4] P. Kumar, Kumar R. “Issues and challenges of load balancing techniques in cloud computing: A survey,” *ACM Computing Surveys (CSUR)*. 2019 Feb 4;51(6):1-35.
- [5] R. Yadav, Zhang W, Kaiwartya O, Singh PR, Elgendi IA, Tian YC. “Adaptive energy-aware algorithms for minimizing energy consumption and SLA violation in cloud computing,” *IEEE Access*. 2018 Oct 2; 6:55923-36.
- [6] N. Subramanian, A. Jeyaraj, “Recent security challenges in cloud computing”, *Comput. Electr. Eng.* 71 (2018) 28–42.
- [7] X. Zhang, T. Wu, M. Chen, T. Wei, J. Zhou, S. Hu, R. Buyya, “Energy-aware virtual machine allocation for cloud with resource reservation,” *J. Syst. Softw.* 147 (2019) 147–161.
- [8] N. Mansouri, B. Mohammad Hasani Zade, M.M. Javidi, “Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory,” *Comput. Ind. Eng.* 130 (2019) 597–633.
- [9] Y.K. Suh, K.Y. Lee, “A survey of simulation provenance systems: modeling, capturing, querying, visualization, and advanced utilization,” *Hum. Centric Comput. Inf. Sci.* 8 (2018) 1–29.
- [10] S.S. Devesh, C. Jinwala, S. Garg, “A survey of simulators for P2P overlay networks with a case study of the P2P tree overlay using an event-driven simulator,” *Eng. Sci. Technol. Int. J.* 20 (2) (2017) 705–720.
- [11] X. Zeng, Garg SK, Strazdins P, Jayaraman PP, Georgakopoulos D, Ranjan R. “IOTSim: A simulator for analysing IoT applications,” *Journal of Systems Architecture*. 2017 Jan 1; 72:93-107.
- [12] S.F. Piraghaj, Dastjerdi AV, Calheiros RN, Buyya R. “ContainerCloudSim: An environment for modeling and simulation of containers in cloud data centers,” *Software: Practice and Experience*. 2017 Apr;47(4):505-21.

- [13] J. Son, Buyya R. "A taxonomy of software-defined networking (SDN)-enabled cloud computing," ACM computing surveys (CSUR). 2018 May 23;51(3):1-36.
- [14] A.R. Arunarani, Manjula D, Sugumaran V. "Task scheduling techniques in cloud computing: A literature survey," Future Generation Computer Systems. 2019 Feb 1; 91:407-15.
- [15] Strumberger, Bacanin N, Tuba M, Tuba E. "Resource scheduling in cloud computing based on a hybridized whale optimization algorithm," Applied Sciences. 2019 Nov 14;9(22): 4893.
- [16] S.E. Chafi, Balboul Y, Mazer S, Fattah M, Bekkali ME, Bernoussi B. "Cloud computing services, models and simulation tools," International Journal of Cloud Computing. 2021;10(5-6):533-47.
- [17] N. Kratzke, Siegfried R. "Towards cloud-native simulations—lessons learned from the front-line of cloud computing," The Journal of Defense Modeling and Simulation. 2021 Jan;18(1):39-58.
- [18] S. Sefati, Mousavinasab M, Zareh Farkhady R. "Load balancing in cloud computing environment using the Grey wolf optimization algorithm based on the reliability: performance evaluation," The Journal of Supercomputing. 2022 Jan;78(1):18-42.
- [19] H. Yu, "Evaluationof cloud computing resource scheduling based on improved optimization algorithm," Complex & Intelligent Systems. 2021Aug;7(4):1817-22

A Hybrid Model Built on VGG16 and Random Forest Algorithm for Land Classification

Mohammed Junaid Ahmed¹, *Ashutosh Satapathy¹, Ch. Raga Madhuri¹, K. Yashwanth Chowdary¹ and A. Naveen Sai¹

¹Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, India

¹mohammedjunaidah@gmail.com, ¹ashutosh.satapathy1990@gmail.com
¹chragamadhuri@vrsiddhartha.ac.in,
¹yashwanthkanaparthi@gmail.com, ¹naveensai.alapati6@gmail.com

Abstract. Working with satellite images using deep learning or machine learning approaches would be exceedingly complex, as each satellite image consists of numerous characteristics. The use of ensemble learning with two different types of machine learning techniques to process images has been a trend in recent times. In this paper, we present a hybrid model that combines the convolutional neural network model (Visual Geometry Group 16) with an ensemble classifier such as the Random Forest Classifier. Our hybrid approach categorizes the Sentinel-2 satellite-collected EuroSAT dataset. Several data augmentation techniques are used to process these images. In order to classify the terrain, characteristics from the spatially linked picture are extracted using the VGG16 model. The RF classifier categorizes the images once the features have been passed on to the instantiated RF classifier. The implementation of the hybrid model has achieved an accuracy of 94.88%, with the major advantage of requiring less time to train and test the EuroSAT dataset.

Keywords: Remote Sensing Data, Convolution Neural Network, Visual Geometry Group 16, Random Forest, Sentinel-2A EuroSAT Images.

1 Introduction

Satellite images are one of the most powerful and important tools which are used by meteorologists to detect the changes on and over the surface of the Earth. These images reassure forecasters about the behavior of the atmosphere as they give a clear, concise, and accurate representation of how events are unfolding. The land cover describes about the various features on the Earth's surface. Land cover classification techniques have been used to identify coastal, agricultural, forest, and urban areas; these techniques are very helpful for the government organizations to obtain information about inaccessible remote regions. For example, in the past few years Land classification techniques have been used to understand the agricultural activities over the regions of Brazil in 2003 and urban expansions in Bangladesh. This research examines the study of classifying land cover which is one of the promising and required remote sensing subjects [1]. The socioeconomic purpose of a piece of land is described by its land use.

The study of land cover and land use images has arisen due to the significant impact that human occupancy has on the environment, which mainly focuses on land use and changes in the land cover. Applications for precise land cover maps

include resource management, environmental monitoring, and the planning of diverse operations over the existing or projected land cover. Large land cover maps required to identify, track human activity, and map land biogeography are being captured using satellites such as Sentinel-1, Sentinel-2, and LandSat [2, 3]. The availability of satellite data has driven the development of novel methods to map land cover using remotely sensed data. Data from the Sentinel-2 satellite have been frequently used to calculate urban changes, agricultural changes, and terrestrial changes. Sentinel-2 images have been available since 1972, making them ideal for mapping the land cover. Numerous automated and semi-automatic categorization approaches have recently been developed, yielding precise results for land cover mapping [4]. An essential approach in remote sensing analysis is image classification, which distinguishes pixels based on their contribution to land cover. However, classifying land cover images has always been difficult, and gathering training data is quite expensive and complicated. Image preprocessing techniques have to be more concentrated on the geometrical, textural, and contextual properties surrounding each pixel, but at the same time, classifying land cover images at the pixelized level is a hefty task [5].

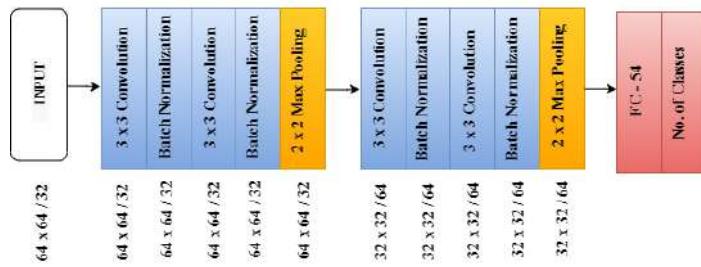


Figure 1. Architeture of a CNN

This article offers a new method for categorizing land use and land changes using Convolutional Neural Network (CNN) models [6]. EuroSat, Sentinel-2 satellite-based images were used as an input dataset in this study. EuroSat is the UK's largest satellite and aerial equipment distributor to the trade, with eight regional branches [7]. In order to determine the land cover, we have used VGG16 to extract the features, and these features have been passed to an instantiated Random Forest (RF) for classification [8]. The suggested land classification hybrid model's scope includes classifying data on land use types, environmental implications, and possible land use alternatives. It also contains data on the type of human habitation in land use. Land use classification describes the useful potential land alternatives, including land cover and land changes, natural resource management, types of human habitation involved in land use, environmental impact assessment, and evaluation of urban expansions. A CNN model automatically retrieves image features by altering the convolutional layer and pooling layer parameters. Certain machine learning models, such as K Nearest Neighbors (k-NN) and logistic regression, are not typically appropriate for applications with large datasets and many variables, whereas CNN models can be used for large and complex datasets. Convolution, batch normalization, pooling, fully connected layers are the four main layers in a CNN, as shown in Figure 1.

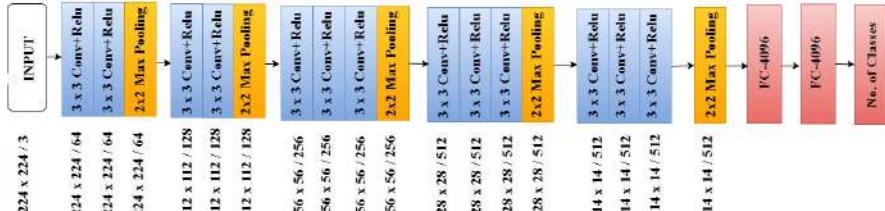


Figure 2. VGG16 architecture

At the University of Oxford, K. Simonyan and A. Zisserman implemented one of the popular CNN models [9], VGG-16, whose error rate was 8.1%, and placed it as one of the top five classifiers in ILSVRC 2014 [10]. Figure 2 shows the whole network consists of 13 convolution layers of different stack sizes (two $224 \times 224 / 64$, two $112 \times 112 / 128$, three $56 \times 56 / 256$, three $28 \times 28 / 512$, and three $14 \times 14 / 512$) and three fully connected layers. It uses an RGB image of size $224 \times 224 / 3$ as input to generate the class label at its final layer. The max pooling layer reduces the size of feature maps after the second, fourth, seventh, tenth, and thirteenth convolutional layers, as well as the ReLU activation function, are used with the first 15 layers to filter out required features while reducing computational cost. The Softmax function after the last layer computes the probable value for each class, while a 0.5 dropout rate reduces the issues of overfitting or underfitting and provides the finest results to the Softmax function for classification. At last, a stochastic gradient descent optimizer with a learning rate tunes the network's weights by minimizing the classification error rate in every epoch as the training proceeds. On the other hand, it requires a huge amount of computational resources to train 138 million trainable parameters. Features from the last pooling layer of VGG16 are taken out and given to the RF algorithm for land classification. This not only cuts down on training time, but also on the overall cost of computing. In the next section, we will discuss different algorithms or models that were suggested by different authors and tested on different standard datasets.

2 Related Work

Jacob et al. proposed a supervised method using the RF algorithm for classifying land use and land change that detects changes in two stages [11]. This study examines the interferometric mapping of the Sentinel-1 satellite for vegetation and land use. The research areas in the European regions were selected for the 2015–2016 campaigns to maximize the range of land cover. The abovementioned strategy trained on the Sentinel-1 image intensities and the smallest temporal baseline features from 9 to 15 classes, which achieved an overall training accuracy of between 70% and 90%.

Jalayer et al. focused on the Doana Wetlands National Park in Spain, a flat region, for land use and land cover change using multitemporal Landsat imagery [12]. It aimed at identifying temporal and geographical modifications in the land use and land cover change patterns of the Chalus watershed region during the last two decades and forecasting the land use and land cover modifications along with

the patterns in the Chalus watershed in 2040. Three Landsat images were used from the ETM, and 200 sensors have been employed in this work. To compare closely the land use and land cover change maps for 2021 and 2040, the support vector machine method and another model, the Markov chain, produced the transition maps and plotted the probability matrices between the land use and land cover change types. The success rate of this LULC research project was 90%.

G'omez and Meoni had demonstrated MSMatch, the first semi-supervised learning algorithm for scene categorization [13]. It appeared to be the best when compared with other supervised methods on the benchmark land use datasets from EuroSAT and UC Merced. The key contributions of this study are the extension of current developments in nearest neighbor designs and the use of semi-supervised learning techniques to multi-spectral and remote sensing. The analysis of crucial pipeline elements through ablation studies focuses on the discovery of heterogeneous characteristics of EuroSAT and UCM classes. In terms of robustness against Gaussian noise with spectral distortions, this approach was trained on EuroSAT and UC Merced land use datasets. On these datasets, a trained nearest neighbor algorithm outperforms the past techniques by up to 19.76% and 5.59%, respectively. This work has achieved 90.71% and 95.86% accuracy on the UC Merced land use dataset and EuroSAT, respectively. using just five image samples per class.

Jain et al. proposed a study that offers "Remote Sensing Bootstrap Your Own Latent" (RSBYOL), which improves the classification techniques in the remote sensing areas where the data are non-trivially different from natural RGB images [14]. However, the remote sensing community is plagued by two significant issues: multi-sensor and the nature of multimodal data acquired from several satellites, and labeled data scarcity. In contrast to conventional RGB pre-trained models, they have examined the values of pre-trained models with RSBYOL. They had done this by contrasting their own RSBYOL weights with well-known ImageNet pre-trained weights. They have trained the RSBYOL using Multi-Spectral Synthetic Aperture Radar (MS-SAR) and multispectral-only data, demonstrating the value of learning invariant characteristics from MS-SAR data. This work showed that using MS and SAR as different ways to look at a network for training was better than just using MS data, with an accuracy of 98.30%.

Liu Z. et al. had proposed a model that built on a dual-channel network for classifying the land cover using an encoder-decoder architecture [15]. The authors have acquired a new dataset with 30,000 pictures and a spatial resolution of 256x256 for grid sampling and data augmentation. Using a random division approach, the dataset has been split into two parts: training data consisting of 22,500 images and test data consisting of 7,500 images. The existing dataset was improved and cropped to create a custom dataset with a spatial dimension of 256x256, having 9600 images for training and 2400 images for testing. The overall accuracy of the Deep Fully Connected Neural Network (D-FCN) increased to around 6.67% compared to other models, and the Kappa coefficient rose by 0.0944, 0.0534, 0.0576, and 0.0753, respectively. Locally speaking, the D-FCN has a significantly better extraction effect on these categories, resulting in precision values of 89.06% and 85.55%.

Pleiades et al. had implemented an advanced CNN model in conjunction with Object-based Image Analysis (OBIA) to map the land use and land classification over the coastal region of Ain Témouchent, western Algeria [16]. At first, training and test samples were segregated, and spectral features were generated from these samples. The proposed model was based on a combined approach of CNN deep modeling with OBIA for land use and land classification. Land use and land classification utilized OBIA and pixel-based techniques using RF and SVM classifiers, and the accuracy of land use and land classification maps was evaluated. For both OBIA and pixel-based techniques, the results obtained with RF were significantly better than those obtained with SVM. RF outperformed SVM regardless of the method used. RF-OBIA and SVM-OBIA obtained 91% and 72% success rates, respectively, whereas RF-Pixel and SVM-Pixel obtained 80.1% and 77.4%, respectively.

Kang et al. had implemented the land change detection system, which includes the use of the complementary properties of the two sensors [17]. They presented a novel modular and fully convolutional network for improving land cover classification accuracy by fully utilizing the complementary features of the two sensors. They expected joint classification using optical and SAR images to achieve higher accuracy than the classification models using single-sensor images. For this purpose, they designed a modular model that consists of three modules: an encoder, a multiscale module, and a decoder. These modules could be replaced with any other modules, and fusion approaches could be changed. As a result, the accuracy and precision scores were 80.85% and 84.38%, respectively.

Busquier et al. had proposed an approach that focuses on two types of mapping, namely land cover and crop vegetation [18]. These classification studies were conducted using a series of images obtained by TanDEM-X Sentinel-1 using the C band and EuroSAT using the L-band. The classification study's primary objective was to investigate how well feature and band combinations perform when used to categorize data. In this regard, these frequency bands can be used individually or simultaneously. The RF algorithm performs the categorization process. It was a cutting-edge supervised classifier with a solid reputation for performance. When only one frequency band was used, L-band had the best overall accuracy of 81%, followed closely by C-band, while X-band only had 70%. The accuracy of the land cover categorization using 20 intensity pictures in the C- and X-bands is 61.49% and 77.62%, respectively. The C-band had the most significant classification accuracy of the three bands when used with all intensity channels in all bands.

3 Proposed Methodology

Figure 3 depicts the work flow of our proposed model. Each image is 64x64 pixels in size. Four class labels are one hot encoded and are stored in respective lists. Once the preprocessing is done, the images are passed on to the VGG16 model. The VGG16 model had already been trained using the ImageNet dataset, which consists of 14 million images categorized into 10,000 class labels [19]. The VGG16 model is used to only extract the features of the images, and

approximately 269 characteristics are retrieved from these 12000 pictures. Once the VGG16 model has extracted the features, an RF classifier is instantiated, and these extracted features are trained against the one-hot encoded class labels. Once the classifier has been trained, the model is prepared for the testing step. During the testing phase, approximately 2400 images are used as testing data.

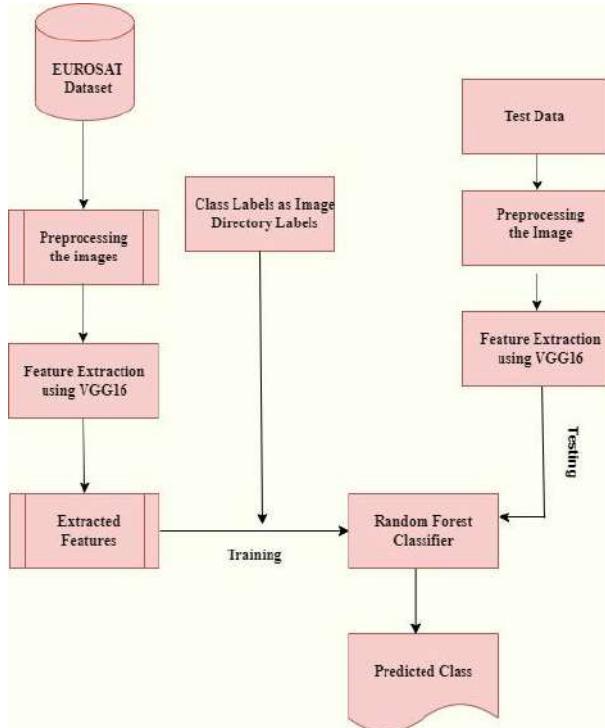


Figure 3. Process Flow Diagram of land classification

The data processing stage prepares the raw data for training the ML or DL models. Before building a deep learning model, preprocessing the data is crucial. One-hot encoding is a critical stage in improving prediction accuracy; it is the process of changing the categorical data variables to be fed to ML and DL models. The categorical features for DL models are processed frequently using one hot encoder. This process generates binary numbered values, which can categorize each class label. A total of four categories of class label have been one-hot encoded. The VGG16 input layer, which takes in images of size 224x224 pixels with three channels, was developed based on the AlexNet model. Around 14 million images from the ImageNet dataset, consisting of 10,000 categories, were used to train the VGG-16 model, whose trainable parameters were fine-tuned at each layer [20]. 269 features are extracted from each image from the EuroSAT dataset during the training phase. The extracted features are used to train the RF classifier once it has been initialized with 50 trees. The hybrid model currently consists of 13 convolutional layers of VGG-16 and RF, where the RF classifier is substituted for three fully connected layers of the naive VGG-16 model. Training and validating a random forest classifier with fewer features take less time than

training and validating a fully connected neural network with 4096 neurons with a larger number of features.

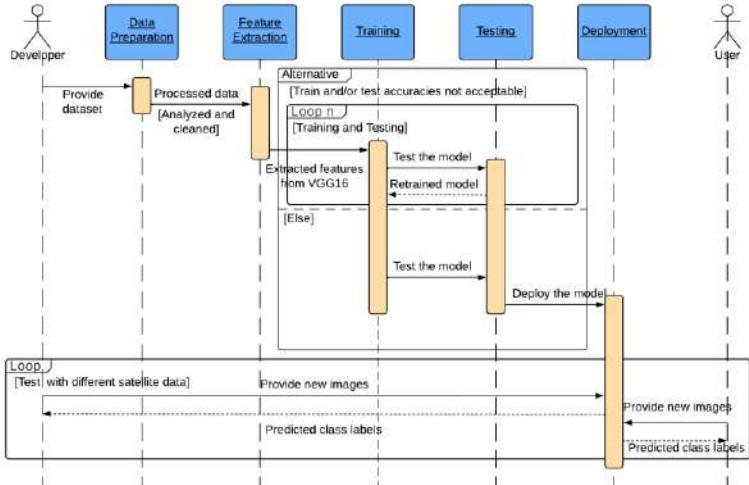


Figure 4. Sequence diagram of land classification system

Figure 4 shows the sequence diagram of the land classification system, in which the developer gathers the satellite data, pre-processes it using Scikit modules, and then extracts the features using the pre-trained weights of the VGG16 model. These characteristics are assigned to the RF Classifier, which is used in place of the VGG-16 fully connected dense layers. This model is evaluated against the images when it has been fully trained. When the level of accuracy is acceptable, the model is deployed and made available for the user to categorize the photos. If the accuracy is unsatisfactory, the model is retrained by adjusting hyper-parameter values, and the procedure is repeated until the accuracy is satisfactory. The architecture of the proposed hybrid model is shown in Figure 5. Deep Learning algorithms are widely used in image and video recognition systems since they have the capability of automatically generating features from the time series data and frequency representation images. These try to learn high-level features from data in an incremental manner. This diagram shows that the features extracted using VGG16 are given to RF for classification into different classes.

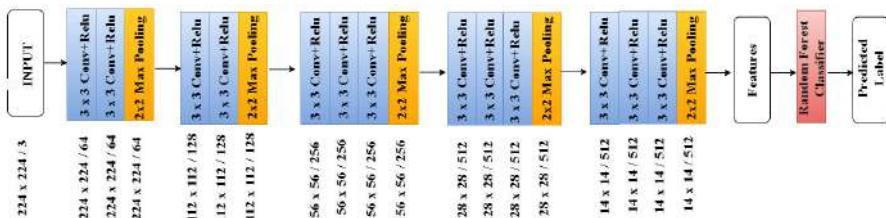


Figure 5. Architecture of the proposed hybrid model VGG16–RF.

In the above figure, the architecture diagram consists of 13 layers of the VGG-16 model. The convolutional layers are the fundamental components of VGG16,

which typically consist of a combination of linear and non-linear operations such as convolutional operations and activation functions. The Random Forest algorithm is one of the most widely used image classification algorithms. Among all radios, RF has the highest accuracy and can handle massive amounts of data with thousands of variables. As we have used a huge dataset in this research, it is obvious that it can lead to an overfitting issue; thus, using an algorithm that can handle this issue on its own would be the most appropriate step. The RF classifier doesn't face the overfitting issue because it takes the average of all predictions, canceling out the biases and thus fixing the overfitting problem. Features are obtained using these 13 layers, which act as a feature extractor in the VGG-16 model. The extracted features are used to train an RF classifier against the one hot encoded class labels.

4 Results and Analysis

EuroSAT dataset contains 12,000 images of 4 different types of land. Figure 6 shows sample land images from the dataset. Each picture is 64x64 and occupies a 100-meter square, with a cloud cover of 1.505%. A total of 13 bands were used in the Sentinel-2 satellite.

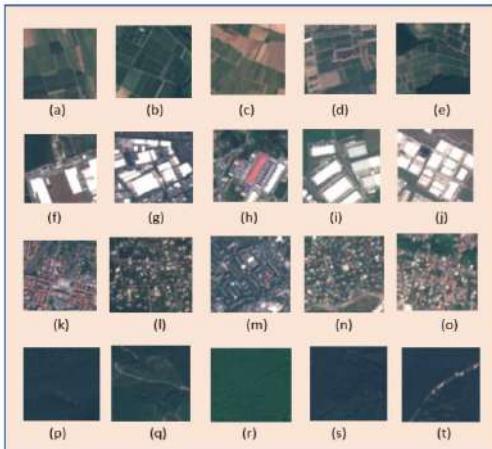


Figure 6: Sample images from EuroSAT dataset: (a-e) crop land; (f-j) industrial land; (k-o) residential land; (p-t) forest land.

Figure 6 represents 4 different categories of images; each category consists of 4000 images, for a total of 12000 images. These have been used in our land classification hybrid model. In the above figure, images from (a-e), (f-j), (k-o), and (p-t) belong to crop land, industrial land, residential land, and forest land, respectively. Table 1 depicts the precision, recall, F-score, and accuracy of the proposed hybrid model trained on the EuroSAT dataset. Precision is the total number of correctly classified image samples to their actual classes divided by the sum of all correctly classified image samples to their actual classes and the sum of

samples of different classes are incorrectly classified to each class. This measure of precision is shown in equation 1.

$$\text{Precision} = \frac{\sum_{i=1}^4 \text{TP}_i}{\sum_{i=1}^4 \text{TP}_i + \sum_{i=1}^4 \text{FP}_i} \quad (1)$$

Recall is the total number of correctly classified image samples for their actual classes, divided by the sum of all correctly classified image samples for their actual classes and the sum of image samples of each class incorrectly classified for other classes. This measure of recall is shown in equation 2.

$$\text{Recall} = \frac{\sum_{i=1}^4 \text{TP}_i}{\sum_{i=1}^4 \text{TP}_i + \sum_{i=1}^4 \text{FN}_i} \quad (2)$$

The F-score, also known as the F1-score or F-measure, is used to assess the performance of a machine learning model using the precision and recall scores, calculated using the formula shown in equation 3. Its range is between 0 and 1, and a higher value indicates better performance. It is used to balance the precision and recall score and will be useful when the number of image samples per class is inadequate.

$$\text{F-Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

Table 1. Performance score of the proposed hybrid model on the EuroSAT dataset.

Metric	Score
Precision	0.940990407
Recall	0.940833333
F Score	0.940820619
Accuracy	0.9488

VGG16 uses a stochastic gradient descent optimization function with batch size, momentum, and weight decay set to 128, 0.9, and 0.0005, respectively, and all the layers use an equal learning rate of 0.001. Table 1 explains the precision score, recall score, and F score achieved by our hybrid model, whose precision, recall and F score are 0.94099407, 0.940833333 and 0.940820619, respectively.



Figure 7: Output images classified by the system: (a, b) crop land, (c, d) industrial land (e, f) residential land, (g, h) forest land.

Figure 7 shows the outputs that are classified by our hybrid model when these eight images are randomly given as an input to the system. Each piece of land image in the above figure is given as an input to our hybrid model, and the subsequent classified label has been described in the image caption. For example, (a, b), (c, d), (e, f), and (g, h) have been accurately classified as crop land, industrial land, residential land, and forest land. The model's successes and failures can be plotted using a confusion matrix. Its rows represent a predicted class, and each column of the matrix represents an actual class. For each class, all diagonal cell values are true positive, whereas cells in black along each column and row are false negative and false positive for each class, respectively.

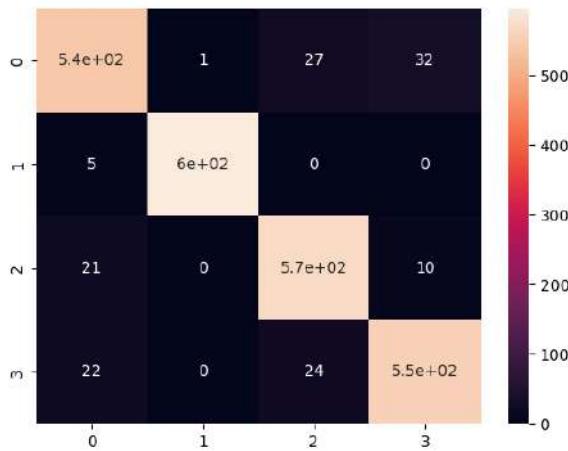


Figure 8: Confusion Matrix of the proposed hybrid model on the EuroSAT dataset.

Figure 8 shows how the proposed hybrid model accurately predicted all four classes. In the above figure, the (0, 0), (1, 1), (2, 2), (3, 3) cells in the matrix represent a total of 5.4E+2, 6E+02, 5.7E+02, and 5.5E+02 images that have been accurately classified as crop land, industrial land, residential land, and forest land. A small percentage of images were not correctly classified in all four classes, which has been reflected in the model's accuracy decreasing by 5.12% in table 1.

Table 2. Comparison between accuracies of proposed model with respect to previous work.

S. No.	Literature	Classifier	Accuracy
1	Jacob et al. [11]	RF	92.4%
2	Jalayer et al. [12]	SVM	77.62%
3	G'omez and Meoni [13]	CNN	70-90%
4	Jain et al. [14]	ResNet	90.2%
5	Liu et al. [15]	D-FCN	95.7%
6	Zaabbar et al. [16]	SVM	99.30%
7	Kang et al. [17]	CFNet	85.55%
8	Busquier et al. [18]	RF	80.1%

9	Proposed hybrid model	VGG16-RF	94.88%
---	-----------------------	----------	--------

Table 2 presents a comparison between the previous work that have been executed on the same dataset. Each model was either fully CNN-based or a traditional machine learning classification model, which had shown relatively greater accuracy. The work published by Gomez and Meoni achieved accuracy in the range of 70–90% [13]. Kang and Xiang achieved an accuracy of 85.5% [16]. On the other hand, Busquier achieved an accuracy of 80.1%. Few of the published work had worked with RGB images, and even fewer had worked with the spectral bands of the dataset and obtained decent accuracy [17]. We can clearly see from the above table that D-FCN and SVM models achieved greater accuracy when compared to only CNN-based models. In this work, our hybrid model using the VGG16-RF classifier has achieved an accuracy of 94.88%.

4 Conclusion

In this study, we discussed how difficult it is to categorize land cover and its uses over time. The EuroSAT dataset, acquired using the Sentinel-2 satellite, has been used for the proposed system. In all, 12,000 annotated images, categorized into four different classes covering 13 different spectral bands, are included in the training and test sets. We utilized the most highly trained model, VGG16, to extract 269 features from each image sample in this dataset and supplied them to the RF algorithm for training and testing purposes. Aside from that, we then evaluated the proposed hybrid model using remote sensing datasets and compared the results to methods that had previously been reported. The proposed model is built on the ensemble learning of CNN and machine learning classifiers. The proposed study could lead to a number of useful results, such as changes in land cover and use, urban growth, monitoring agricultural activities, and so on.

The result shows that the custom hybrid DL model performed significantly better than other ML models, and its performance on a four-class dataset is 94.88%. When the model is used in real-time applications, it will not only help us find out what's going on in agricultural, industrial, forest, and residential areas, but it will also help us find problems in dangerous or hard-to-reach places and manage and keep an eye on them without having to drive there. In the future, we plan to validate the performance of the proposed model on large-scale datasets and cross-sample datasets. To meet current demands, extensive work must be done to develop sophisticated, lightweight deep learning models that incorporate a wide range of features for land use and land classification with greater accuracy and minimize the computation time.

References

- Chang, Y. et al. (2018, February). Review of land use and land cover change research progress. In IOP Conference Series: Earth and Environmental Science (Vol. 113, No. 1, p. 012087). IOP Publishing.
- Cihlar, J. (2000). Land cover mapping of large areas from satellites: status and research

priorities. International journal of remote sensing, 21(6-7), 1093-1114.

3. Drusch, M. et al. (2012). Sentinel-2: ESA's optical high-resolution mission for GMES operational services. Remote sensing of Environment, 120, 25-36.
4. Jiang, D. et al. (2012). A simple semi-automatic approach for land cover classification from multispectral remote sensing imagery. e45889.
5. Deepthi, S., Sandeep, K., & Suresh, L. (2021). Detection and Classification of Objects in Satellite Images using Custom CNN. International Journal of Engineering Research & Technology, 10(6), 629-635.
6. Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017, August). Understanding of a convolutional neural network. In 2017 international conference on engineering and technology (ICET) (pp. 1-6). Ieee.
7. Helber, P. et al. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12(7), 2217-2226.
8. Pal, M. (2005). Random forest classifier for remote sensing classification. International journal of remote sensing, 26(1), 217-222.
9. Mascarenhas, S., & Agarwal, M. (2021, November). A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification. In 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON) (Vol. 1, pp. 96-99). IEEE.
10. ILSVRC2014 Results. (2014). ILSVRC2014 Results. Retrieved January 24, 2023, from <https://image-net.org/challenges/LSVRC/2014/results>.
11. Jacob, A. W. et al. (2020). Sentinel-1 InSAR coherence for land cover mapping: A comparison of multiple feature-based classifiers. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13, 535-552.
12. Jalayer, S. et al. (2022). Modeling and predicting land use land cover spatiotemporal changes: A case study in chalus watershed, Iran. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 5496-5513.
13. Gómez, P., & Meoni, G. (2021). MSMatch: Semisupervised Multispectral Scene Classification With Few Labels. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 11643-11654.
14. Jain, P., Schoen-Phelan, B., & Ross, R. (2022). Self-Supervised Learning for Invariant Representations from Multi-Spectral and SAR Images. arXiv preprint arXiv:2205.02049.
15. Liu, Z. et al. (2022). A Dual-Channel Fully Convolutional Network for Land Cover Classification Using Multifeature Information. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 2099-2109.
16. Zaabar, N., Niculescu, S., & Kamel, M. M. (2022). Application of convolutional neural networks with object-based image analysis for land cover and land use mapping in coastal areas: A case study in Ain Témouchent, Algeria. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 5177-5189.
17. Kang, W., Xiang, Y., Wang, F., & You, H. (2022). CFNet: A Cross Fusion Network for Joint Land Cover Classification Using Optical and SAR Images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 1562-1574.
18. Busquier, M., Lopez-Sanchez, J. M., Ticconi, F., & Flouri, N. (2022). Combination of Time Series of L-, C-, and X-Band SAR Images for Land Cover and Crop Classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15, 8266-8286.
19. Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). IEEE.
20. Team, K. Keras documentation: VGG16 and VGG19. *Keras. io*. Retrieved January 24, 2023, from <https://keras.io/api/applications/vgg/>

Facial Expression Recognition using Transfer Learning with RESNET50

Shantala S Hiremath¹, Jayaprada Hiremath², Vaishnavi V Kulkarni³, Harshit B C⁴,
Sujith Kumar⁵ and *Mrutyunjaya S Hiremath⁶

¹Specialist in Image Processing, Sony India Software Centre, Pvt Ltd. Bangalore -
560103, Karnataka, India
¹shantala.hiremath@gmail.com

²Research Scholar, Department of Computer Science and Engineering,
Visvesvaraya Technological University, Belgaum – 590018, India

³Assistant Professor, Department of CSE, Alvas Institute of Engineering and
Technology, Mijar, Moodbidri - 574225, Karnataka India

⁴Senior Developer, Department of Image Processing, eMath Technology, Pvt. Ltd.
Bangalore - 560072, Karnataka, India

⁵Principal Engineer (Artificial Intelligence and Machine Learning) VVDN
Technologies Pvt. Ltd. Bangalore - 560066, Karnataka

⁶Department of Image Processing, eMath Technology, Pvt. Ltd. Bangalore -
560072, Karnataka, India
⁶*mrutyunjaya.publications@gmail.com

Abstract. Facial expression recognition mimics human coding abilities and delivers non-verbal human-robot communication cues. Machine learning and deep learning techniques enable real-world computer vision applications. Deep learning-based Facial Emotion Recognition models have under-fitted or over-fitted due to inadequate training data. They are using FER2013's 7 picture categories. Face detection using Adaboost, scaling with OpenCV, and contrast improvement with Histogram equalization preprocess these pictures. These pre-processed pictures are given to the ResNet50 pre-trained Network, which obtained 77.3% accuracy. Transfer learning improves this outcome. By running pre-processed pictures through ResNet50's FC1000 layer, features are retrieved and trained using a Multiclass Non-linear Support Vector Machine (SVM) classifier with seven classes. Training with 89.923% accuracy creates a knowledge base. Emotion recognition techniques let robots understand people, which can improve HCI.

Keywords: Expression recognition, transfer learning, histogram equalization, multiclass classification, support vector machine, resnet50, deep learning.

1 Introduction

Emotions are not stored [1]. Computers cannot recognize facial expressions [2]. Studies [3] show that facial expressions can identify simple and complicated emotions. Primary emotions are anger, contempt, fear, pleasure, sadness, and surprise. No one needs to be taught the six fundamental emotions. Complex emotions are not innate; everyone has them. Beliefs and morals produce complicated emotions. Emotions reveal the mind, attitude, and behavior. Emotion-reading machines improve communication. Emotion detection can detect interrogator lies and enhance gameplay. In education, it can measure student engagement [4].

Deep learning surpasses DIY methods, studies suggest. Computer vision leverages deep learning [5]. Deep network training is time- and data-intensive. Rare are large datasets and robust systems. A neural network's architecture resembles a brain. Pretrained on millions of images, it can do several tasks. Data and actions are first-layer features. CNN layers gather abstract visual information to build a noteworthy feature. Customize data-collection layers. GoogleNet, ResNet, VGGNet [6], and DeepNet are pre-trained. Using the dataset, these networks may provide task-specific outputs.

A trained network is needed for CNN image categorization. Test accuracy lowers with non-network-class photos. Moreover, retrain the network. This requires millions of images and a strong GPU [7]—transfer learning. Transfer learning impacts fully-linked layers. Learned layer weights are entirely related. SVM transfers flattened convolutional basis values. A pre-trained network classified dataset photos into seven emotions. Second, a hybrid technique collects information from a pre-trained network's FC1000 layer and trains a multiclass non-linear SVM classifier. Experiments show that the hybrid method works—Block diagram in Figure 1.

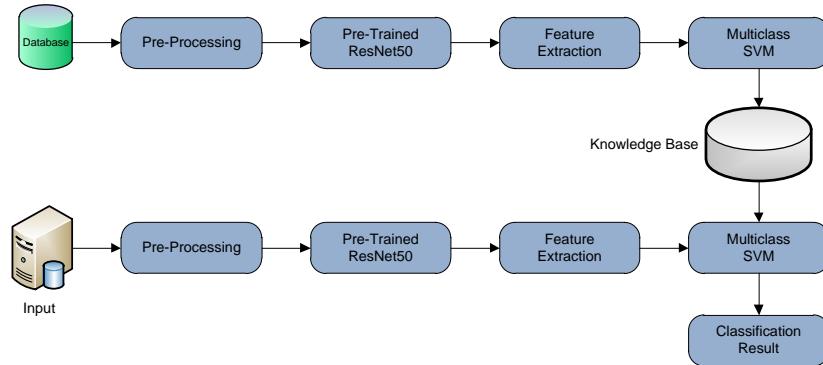


Fig. 1. Block Diagram

2 Literature Survey

CNN's need many data to train. Using millions of photos, researchers trained CNN for Transfer Learning. CNN-trained AlexNet Research suggests transfer learning alone might be sluggish. Time-saving SVM SVM classified AlexNet picture characteristics. Custom CNNs stink. Visible and thermal imaging was tested. NVIE has a 99.3% accurate thermal database. CK+'s visual accuracy was 98.3% [1].

Two expression-recognition methods were developed. Classify AlexNet using transfer learning and VGG19. SVM identified AlexNet and Vgg19 features. Techniques JAFFE-tested. The proposed system matches the best existing system's 94% accuracy. Small datasets are key. CNN-Deep shines. CNN learns SVM. VGG19-SVM cascade is 94% accurate. Training four JAFFE models. Improved JAFFE Data Set [7].

This article improves FaceLiveNet's emotions on the FER2013 model and achieved 70%. Transfer learning enhances test accuracy by 12.9% using the FER2013 basic emotion recognition model. The rebuilt model learns map outputs from eyebrows, chins, and closed eyelids. Integrating image data with the original learning emotion database boosted the model's accuracy [8].

This article discusses EmotiW 2017's group-level emotion challenge. We got face feature vectors from a CNN trained for face recognition, not emotion. An emotion-predicting Random Forest classifier ensemble was trained. The method employs Viola-Jones cascades, HOG features, facial landmarks, and a deep CNN trained for face recognition to identify faces in a group picture and compute their median attributes. RFs determine. Training RFs with vibrant visuals. EmotiW 2017 is not big enough to build a feature extractor. Pipeline data accuracy was 75.4%, 23% over the baseline 78.53 [9].

This article describes 2015's Static Facial Expression Recognition in the Wild sub-challenge. We fine-tuned an ImageNet-trained network incorporating face expression and contest data. Cascaded fine-tuning using linked datasets improves results. The top submission had 48.5% validation and 55.6% test accuracy, vs. 35.96% and 39.13% for baseline. CNN's trained using supplemental facial expression datasets, and EmotiW can enhance accuracy by 16%. Finally, labeling faces with sensitive emotions were complex [10].

This document describes state-of-the-art AI solutions (datasets and algorithms). This study examined detection, extraction, and classification with computing resources. By addressing present and future research difficulties, we concluded that further work is needed, such as FER in 3D face shape models and emotion recognition in obstructed photos. Real-time FER is difficult [11].

Deep learning emotion recognition framework. The recommended strategy speeds up and enhances CNN training with Gabor filters. Gabor filters boost system learning. Gabor filters extract NN subfeatures. This helps CNN detect emotions [12].

This study proposes a real-world emotion detection model. VGG13 has more parameters than LER. FERFIN is quieter and labels better than FERPLUS. This paper presents a lightweight emotion recognition (LER) model using a convolution layer and model compression. DenseNet-3 scored 71.73 percent on FER2013, 0.57 percent more than the first team. FERPLUS' validation set gave DenseNet-2 85.58 percent. DenseNet-3 is 41-times simpler than VGG13. FERFIN's 0.21 million-parameter DenseNet-2 model was 85.89% accurate [13].

This paper presents a CNN-KNN hybrid FER model for Raspberry Pi 4. Fully Connected Layer, Loss Function, Loss Optimizer, Learning Rate, Class Weights, and KNN Distance are optimized. The hybrid CNN-KNN model achieved 75.3% accuracy on Raspberry Pi hardware with limited processing power, memory, and storage, a 0.6% increase over the CNN model, and a 0.1% improvement over state-of-the-art FER models. The preferred model is 0.7526 and 0.9393. We integrated JAFFE, KDEF, and FER-2013 training data to boost the sample size. We compared pre-trained EfficientNet FER models. KNN helps CNN [14].

Continuous-dimension face databases are rare (e.g., valence and arousal). We gathered and recorded spontaneous facial expressions (called AffectNet). Affected identifies 1,000,000 face photos using 1250 emotion-related keywords in six languages. Affected is the most extensive facial expression, valence, and arousal database in the wild. Two baseline DNNs predict valence and arousal. Deep neural networks surpass machine learning and face emotion recognition. Existing facial affect datasets cover one model, a few persons, or emotions. Affected-trained DNN baselines outperformed machine learning [15].

3 Methodology

Figure 1 demonstrates the proposed approach's workflow, including preprocessing, feature extraction from a pre-trained network, and face emotion classification training. Preprocessing eliminates noise and undesirable pixels from photos in three phases (section 3.1). These pre-processed pictures are given to the pre-trained Network, and the FC1000 layer features are retrieved (section 3.2). This technique uses ResNet50, and after extracting features, they are trained using a multi-class non-linear SVM classifier to classify emotions, as detailed in Section 3.3.

3.1 Preprocessing

Preprocessing enhances image quality for better analysis. By preprocessing, we may minimize unwanted distortions and increase application-specific characteristics. Features may vary by application. Face detection, Image resizing, and Contrast improvement utilizing Histogram equalization are explored in this technique. Figure 2 depicts preprocessing.

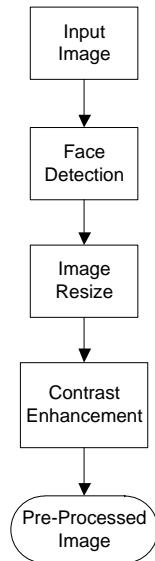


Fig. 2. Preprocessing Flowchart

Face Detection. Face detection is the initial stage in pre-processing; the face is recognized using the AdaBoost classifier and then cropped. Adaboost learns multiple weak classifiers for the training set and combines them into solid classifiers. Combining classifiers improves accuracy. Adaboost classifier employs haar features for classification, as illustrated in figure 3. It can differentiate false positives and true negatives in the data, allowing for a highly accurate model.

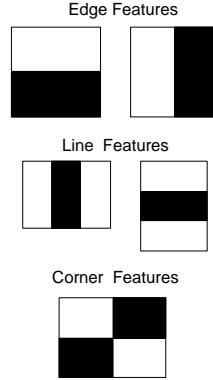


Fig. 3. Haar Features [16]

Haar-Features. Haar wavelets were used in the first real-time face detection. These features compare pixel brightness in adjacent window rectangles by dividing image subsections. These face detection characteristics are eye and cheek rectangles. These rectangles are positioned relative to the target's detection window. The Viola-Jones object identification system drags a target-sized window over the input image and generates Haar-like features for each subsection. This is an object-nonobject threshold.

Resizing. Resizing affects a model's accuracy and training speed. The face picture clipped in the previous stage is enlarged to 224x224 pixels using OpenCV, which is required for the ResNet50 input layer. The interpolation method used to resize an image is INTER LINEAR. Figure 4 and equation 1 demonstrate how INTER LINEAR resizes images using bilinear interpolation.

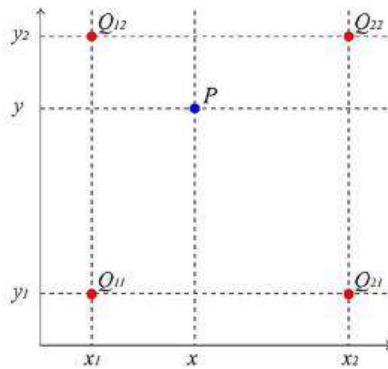


Fig. 4. Bilinear Interpolation [17]

$$f(x, y) = \frac{1}{(x_2 - x_1)(y_2 - y_1)} [x_2 - x \ x - x_1] \begin{bmatrix} f(Q_{11}) & f(Q_{12}) \\ f(Q_{21}) & f(Q_{22}) \end{bmatrix} \begin{bmatrix} y_2 - y \\ y - y_1 \end{bmatrix} \quad (1)$$

where,

x_i, y_i = co-ordinates of pixel

$Q_{11} = (x_1, y_1)$

$Q_{12} = (x_1, y_2)$

$Q_{21} = (x_2, y_1)$

$Q_{22} = (x_2, y_2)$

Contrast Enhancement by Histogram Equalization. Histogram equalization adjusts the contrast to improve visual contrast. Adjusting an image's intensity histogram changes its dynamic range and contrast. Equation 2 describes the equalization of the histogram. Figure 5 shows before-and-after histogram equalization; in figure 5(a), it can be observed that pixels have low contrast, and in figure 5(b), these pixels are moved to high contrast. Table 1 shows the process of face detection and resizing.

$$h(v) = \text{round} \left(\frac{\text{cdf}(v) - \text{cdf}_{\min}}{(M \times N) - \text{cdf}_{\min}} \times (L - 1) \right) \quad (2)$$

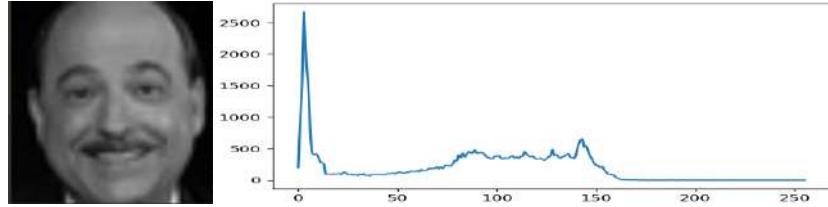
where,

v = value

cdf_{\min} = minimum non-zero value of the cumulative distribution function

$(M \times N)$ = image's number of pixels

L = number of gray levels used



(a)

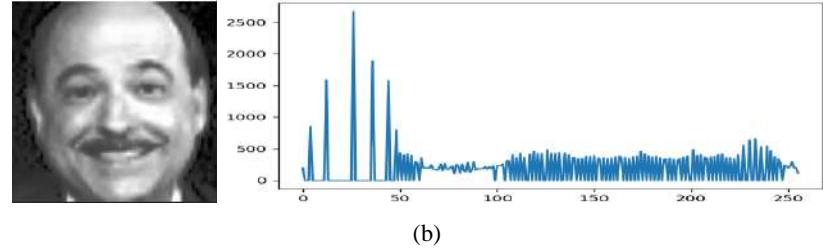


Fig. 5. Contrast Enhancement using Histogram Equalization

Table 1. Face Detection process and Resizing

Input Image	Face Detected Image	Cropped Image	Resized Image
Size 48x48		Size 41x41	Size 224x224
Size 48x48		Size 36x36	Size 224x224

Algorithm of Contrast Enhancement using Histogram Equalization

Inputs: Resized Facial Image

Output: Enhanced Image with Equalized Histograms

Step.1 Consider a discrete grayscale image $\{x\}$ and let n_i be the number of occurrences of gray level i . The Probability of an occurrence of a pixel of level i in the image is

$$p_x(i) = p(x = i) = \frac{n_i}{n}, 0 \leq i \leq L$$

where,

$L = \text{total number of gray levels in the image}$ $n = \text{total number of pixels in the image}$ $p_x(i) = \text{image's histogram for pixel value } i$ <i>Step.2</i> <i>The normalized sum of the histogram is Calculated using the Cumulative distribution function</i> $cdf_x(i) = \sum_{j=0}^i p_x(x=j),$ <i>Step.3</i> <i>The histograms in the input images are equalized using Equation 2</i>

3.2 Feature Extraction

Figure 6 depicts feature extraction. Feature extraction is a type of dimensionality reduction where a considerable number of pixels are efficiently represented to capture exciting elements of the image, which helps determine a person's expression. Pre-trained ResNet50 network from the latest FC1000 layer delivers 2048 features per picture.

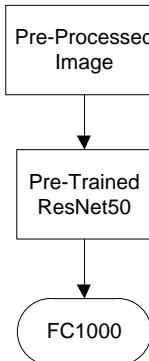


Fig. 6. Process of Feature Extraction

3.3 Training

Figure 7 depicts the suggested categorization flow. Training incorporates pre-trained ResNet50 Network and Multiclass non-linear SVM with seven classes. Transfer learning involves finding parallels between known and unfamiliar information to gain new knowledge. This learning focuses on transferring knowledge between the source and target domains. Figure 8 demonstrates the model's training and validation accuracy.

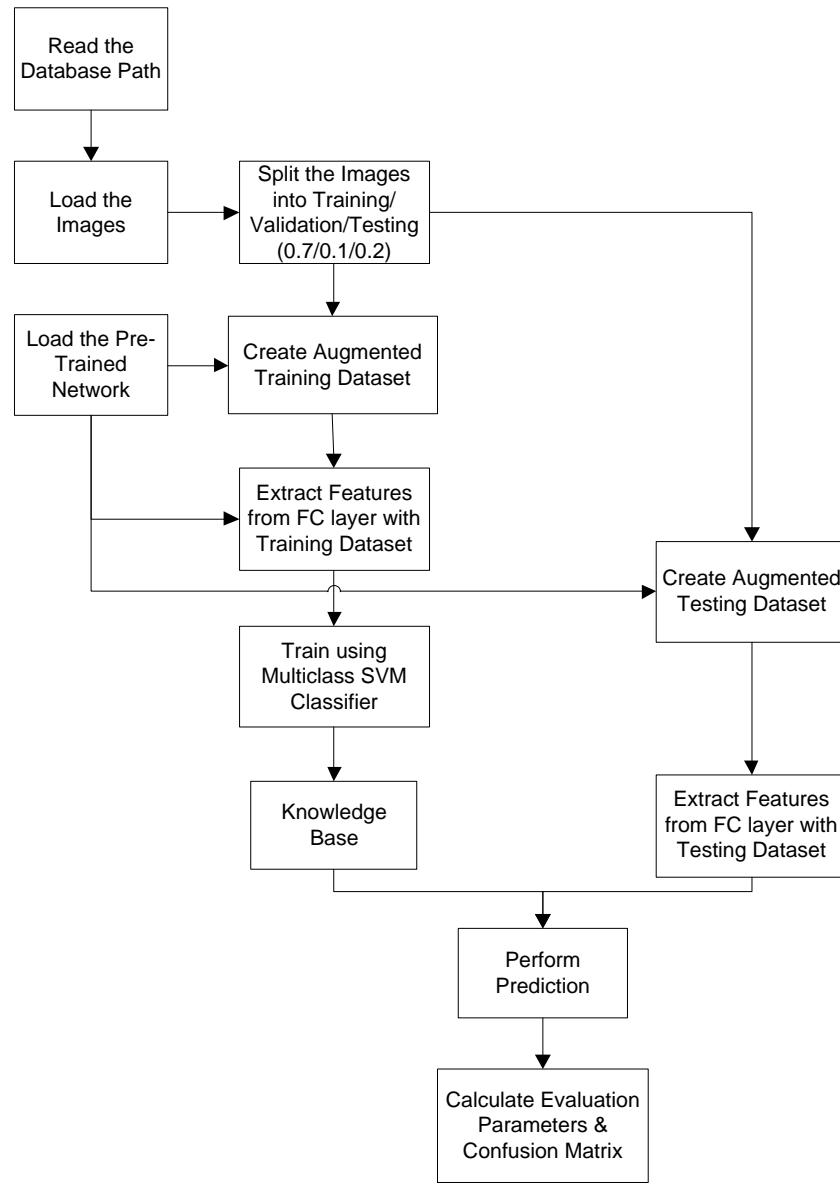


Fig. 7. Classification Flowchart

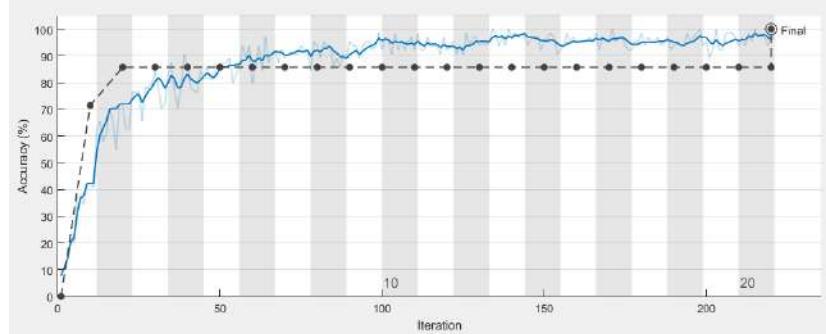


Fig. 8. Training Curve

ResNet50 Network. ResNet50 Network's 49 convolution layers for training are layered into three layers and one fully linked layer. The network receives preprocessed training pictures. Transfer learning is superior. FC1000 layer features from the pre-trained network are used for non-linear SVM classification utilizing RBF (Radial Basis Function) kernel. Figure 9 and Table 2 exhibit ResNet50 details. Equations 3, 4, and 5 demonstrate Convolution, ReLu, and SoftMax.

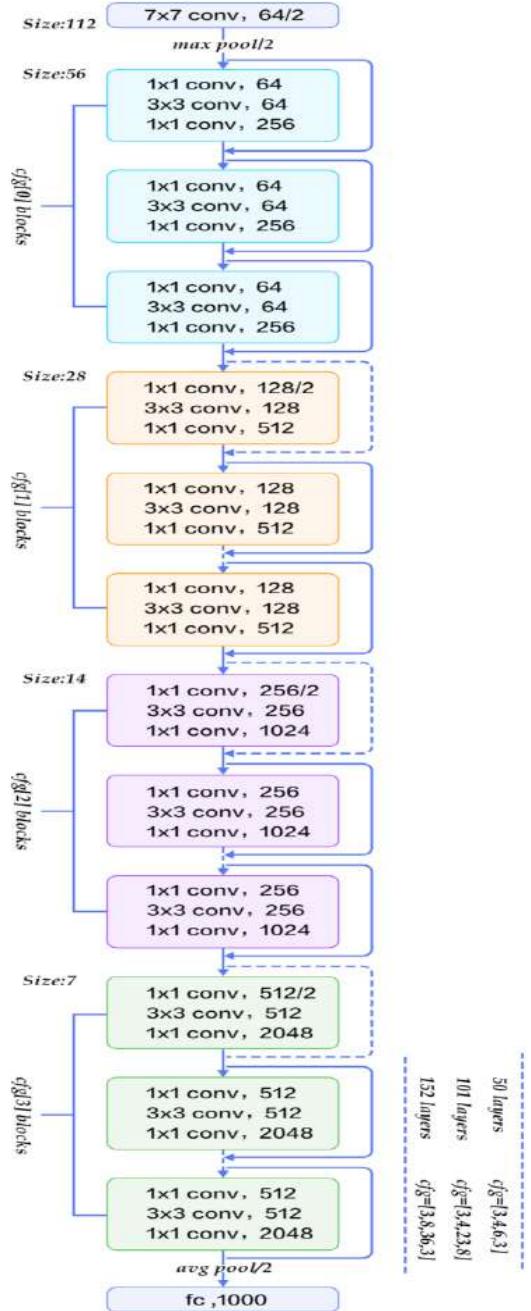


Fig. 9. Architecture of ResNet50

Table 2. ResNet50 Architecture Details

Layer Name	Output size	Layers
Conv1	112x112	7x7, 64, stride 2
Conv2	56x56	3x3 Max pool, stride 2 [1x1, 64 3x3, 64 1x1, 256]x3
Conv3	28x28	[1x1, 128 3x3, 128 1x1, 512]x4
Conv4	14x14	[1x1, 256 3x3, 256 1x1, 1024]x6
Conv5	7x7	[1x1, 512 3x3, 512 1x1, 2048]x3
	1x1	Average Pool 1000-d Fully Connected SoftMax

Equation 3 describes the Convolution layer in a pre-trained network.

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (3)$$

where,

f = signal

g = impulse function

t = time series

τ = deviation

Equation 4 describes the ReLu layer in a pre-trained network.

$$f(x) = \max(0, x) \quad (4)$$

where,

x = input to a neuron

Equation 5 describes the SoftMax layer in a pre-trained network.

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} = \text{ for } i = 1, \dots, K \text{ and } \mathbf{z} = (z_1, \dots, z_K) \in R^K \quad (5)$$

where,

\mathbf{z} = input vector

K = real number

SVM. SVM is a standard supervised classification and regression method. Classifiers like it. It isolates n-dimensional or multidimensional target classes. SVM's primary purpose is to construct the optimum decision boundary (with the most significant margin) to categorize new data points. Multiple lines/decision boundaries may exist in n-dimensional space. We still want the most straightforward data-categorizing decision boundary. SVM hyperplane. Dataset characteristics determine hyperplane dimensions. Margin-maximizing hyperplanes are created. This margin restricts data-point distance. Find a hyperplane that divides n-dimensional data points. The kernel calculates x-n and x-m distances. Datapoints closer together score higher. Figure 10 depicts SVM's kernel.

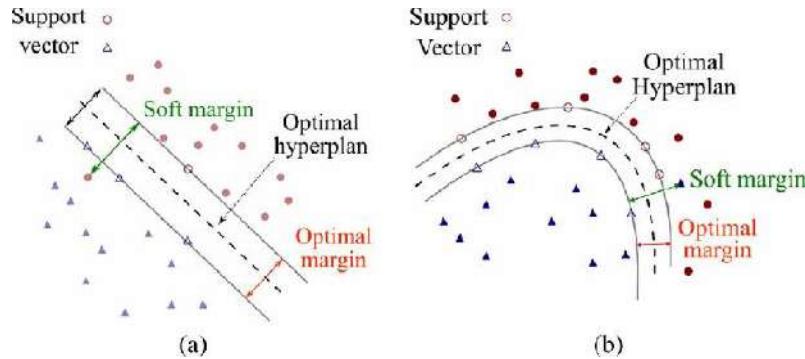


Fig. 10. Linear and Non-Linear SVM

RBF Kernel, similar to K-Nearest Neighborhood Algorithm, is utilized. It benefits K-NN and avoids the space complexity problem by only storing support vectors during training. RBF kernels are employed in several kernelized machine-learning methods like SVM classification. Equation 6 demonstrates RBF kernel math.

$$k(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2) \text{ for } \gamma > 0 \quad (6)$$

where,

\vec{x}_i, \vec{x}_j = feature vectors

$$\gamma = \frac{1}{2\sigma^2}$$

σ = free parameter

Training's knowledge base is developed. The classification performance parameters are measured using test data or pictures. The testing data goes via preprocessing, feature extraction, and Knowledge base.

4 Experimental Results

4.1 Dataset

The data are 48x48 grayscale faces. Faces are automatically registered, so they are centered and occupy the same area in each shot. Figure 11 depicts the 7-class database.

This method uses Kaggle's 7-category dataset. The dataset is 70:10:20 for training, validation, and testing, respectively. All sets have seven category data, including the training set used to train the model, the testing set to assess its accuracy, and the validation set to select and optimize the best model. Table 3 illustrates various database pictures, while Table 4 provides an overview.

Table 3. Database Images

Classes	Images				
Angry					
Disgust					
Fear					
Happy					
Neutral					
Sad					
Surprise					

Table 4. Detailed Overview of Dataset

Classes	Image Size	No. of Images	Training Images	After Augmentation	Testing Images
Angry	48 x 48	3,995	3,196	5772	799
Disgust	48 x 48	436	348	5772	88
Fear	48 x 48	4,097	3,277	5772	820
Happy	48 x 48	7,215	5,772	5772	1,443
Neutral	48 x 48	4,965	3,972	5772	993
Sad	48 x 48	4,830	3,864	5772	966
Surprise	48 x 48	3,171	2,536	5772	635

The aim is to categorize each face by emotion (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set has 28,709 samples, and the test set has 3,589; however, only training set photos were evaluated.

4.2 Experimental Setup

This model has a 10th-generation i-7 CPU, Windows-10 64-bit OS, 6 GB Nvidia GTX, and 8 GB RAM.

4.3 Performance Evaluation

The model's performance is measured by specificity, accuracy, precision, recall, F-1 score, and ROC curve. Specificity, accuracy, precision, recall, and F1 score are expressed mathematically in equations 7, 8, 9, 10, and 11—figure 11 displays the confusion matrix, a classification performance evaluation using expected and actual values. Table 5 shows the parameter measures.

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}} \quad (7)$$

$$\text{Accuracy} = \frac{\text{Total number of Correct Predictions}}{\text{Total number of Predictions}} \quad (8)$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (9)$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (10)$$

$$\text{F1 score} = 2 * \left(\frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \right) \quad (11)$$

Table 5. Performance Parameter Measures

Parameters	Results						
Specificity	98.806						
Accuracy	89.923						
Precision	98.368						
Recall	89.862						
F1-score	93.932						

Confusion Matrix								
True Class	angry	3595	100	25	50	75	50	100
	disgust	3	388	5	7	13	13	7
	fear	55	82	3685	82	69	55	69
	happy	109	145	37	6489	181	145	109
	neutral	131	131	79	27	4465	27	105
	sad	51	77	77	102	153	4344	26
	surprise	91	23	46	91	46	23	2851
		89.1%	41.0%	93.2%	94.8%	89.3%	93.3%	87.3%

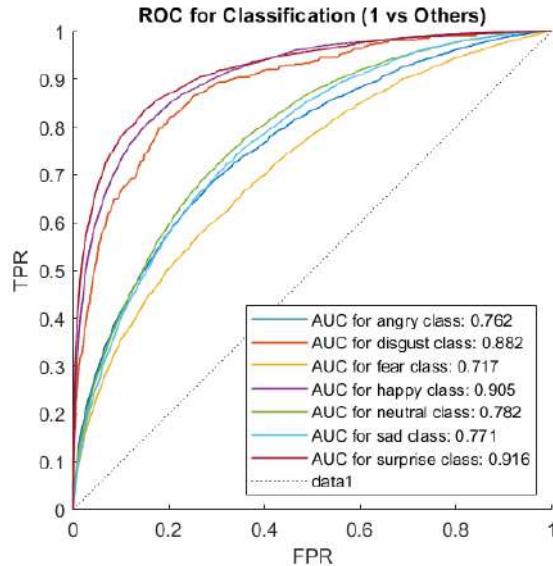
**Fig. 12.** ROC Curve

Figure 12 plots the true positive rate (TPR) against the false positive rate (FPR) to construct the ROC curve. Table 6 provides a comparison with other methods for the FER2013 database.

Table 6. Comparison with other methods for the FER2013 Dataset

Methods	Accuracy
Ensemble Approach [18]	73.73 %
Ensemble Approach [19]	71.236 %
VGG16 [20]	67.2 %
Proposed Approach	89.923 %

5 Conclusion and Future Scope

Comparing detection, extraction, and classification techniques employing transfer learning with a pre-trained ResNet50 network and multiclass non-linear SVM classifier with seven classes yields good results and increased training speed for face emotion identification. After categorization using pre-processed data or

pictures, ResNet50 Network accuracy is 77.3%. After implementing transfer learning by training extracted features from pre-trained ResNet50 Network by passing pre-processed photos to identify the expression using multiclass non-linear SVM algorithm, accuracy increased to 89.923%. Transfer learning increases model correctness. More reliable results can be achieved in the future by using videos and 3D architecture.

References

1. Shaees, Shamoil & Naeem, Muhammad Rashid & Naeem, Hamad & Syed, Hamza & Arslan, Muhammad & Aldabbas, Hamza. (2020). "Facial Emotion Recognition Using Transfer Learning." 10.1109/ICCIT-144147971.2020.9213757.
2. Ramchand Hablani, Narendra Chaudhari and Sanjay Tanwani, "Recognition of Facial Expressions using Local Binary Patterns of Important Facial Parts," International Journal of Image Processing (IJIP) IJIP-738. ISSN (online) 1985-2304, vol.7, Issue 2,2013.
3. Ekman, Paul, Davidson, Richard J, The nature of emotion: Fundamental questions., 1994.
4. X. Cao, Z. Wang, P. Yan, X. Li, "Transfer learning for pedestrian detection," Neurocomp, Vol.100, pp.51-57, 2013.
5. S. Li and W. Deng, "Deep facial expression recognition: A survey," arXiv preprint arXiv:1804.08348, 2018.
6. Swapna, M., Sharma, Dr. Y. K., & Prasad, Dr. B. (2020). "CNN Architectures: Alex Net, Le Net, VGG, Google Net, Res Net." In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 6, pp. 953–959). Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication - BEIESP. <https://doi.org/10.35940/ijrte.f9532.038620>.
7. Hablani, Ramchand. (2020). "Facial Expression Recognition Using Transfer Learning on Deep Convolutional Network." Bioscience Biotechnology Research Communications. 13. 185-188. 10.21786/bbrc/13.14/44.
8. Hung, Jason & Lin, Kuan-Cheng & Lai, Nian-Xiang. (2019). "Recognizing learning emotion based on convolutional neural networks and transfer learning." Applied Soft Computing. 84. 105724. 10.1016/j.asoc.2019.105724.
9. Alexandr Rassadin, Alexey Gruzdev, and Andrey Savchenko. 2017. "Group-level emotion recognition using transfer learning from face identification." In Proceedings of the 19th ACM International Conference on Multimodal Interaction (ICMI '17). Association for Computing Machinery, New York, NY, USA, 544–548. DOI: <https://doi.org/10.1145/3136755.3143007>.
10. Hong-Wei Ng, Viet Dung Nguyen, Vassilios Vonikakis, and Stefan Winkler. 2015. "Deep Learning for Emotion Recognition on Small Datasets using Transfer Learning." In Proceedings of the 2015 ACM on International Conference on Multimodal Interaction (ICMI '15). Association for Computing Machinery, New York, NY, USA, 443–449. DOI: <https://doi.org/10.1145/2818346.2830593>.
11. K. Patel et al., "Facial Sentiment Analysis Using AI Techniques: State-of-the-Art, Taxonomies, and Challenges," in IEEE Access, vol. 8, pp. 90495-90519, 2020, DOI: 10.1109/ACCESS.2020.2993803.
12. M. M. Taghi Zadeh, M. Imani, and B. Majidi, "Fast Facial emotion recognition Using Convolutional Neural Networks and Gabor Filters," 2019 5th Conference on Knowledge-Based Engineering and Innovation (KBEI), 2019, pp. 577-581, DOI: 10.1109/KBEI.2019.8734943.

13. G. Zhao, H. Yang, and M. Yu, "Expression Recognition Method Based on a Lightweight Convolutional Neural Network," in IEEE Access, vol. 8, pp. 38528-38537, 2020, DOI: 10.1109/ACCESS.2020.2964752.
14. M. N. Ab Wahab, A. Nazir, A. T. Zhen Ren, M. H. Mohd Noor, M. F. Akbar, and A. S. A. Mohamed, "Efficientnet-Lite and Hybrid CNN-KNN Implementation for Facial Expression Recognition on Raspberry Pi," in IEEE Access, vol. 9, pp. 134065-134080, 2021, DOI: 10.1109/ACCESS.2021.3113337.
15. A. Mollahosseini, B. Hasani, and M. H. Mahoor, "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild," in IEEE Transactions on Affective Computing, vol. 10, no. 1, pp. 18-31, 1 Jan.-March 2019, DOI: 10.1109/TAFFC.2017.2740923.
16. Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001. Vol. 1. Ieee, 2001.
17. Khosravi, Mohammad R., and Sadegh Samadi. "BL-ALM: A blind scalable edge-guided reconstruction filter for smart environmental monitoring through green IoMT-UAV networks." IEEE Transactions on Green Communications and Networking 5.2 (2021): 727-736.
18. Y. Gan, J. Chen, and L. Xu, "Facial expression recognition boosted by the soft label with a diverse ensemble," Pattern Recognition Letters, vol. 125, pp. 105 – 112, 2019.
19. A. Renda, M. Barsacchi, A. Bechini, and F. Marcelloni, "Comparing ensemble strategies for deep learning: An application to facial expression recognition," Expert Systems with Applications, vol. 136, pp. 1 – 11, 2019.
20. P. Dhankhar, "Resnet-50 and Vgg-16 for recognizing facial emotions," 2019.

Classification of Microorganisms from Sparsely Limited Data using a Proposed Deep Learning Ensemble

Gautam Chettiar¹, Amogh Shukla², Hemprasad Patil³, and Sumit Jindal³

¹ Department of Communication Engineering, Vellore Institute of Technology,
Vellore, Tamil Nadu, 632014

² Department of Software Systems, Vellore Institute of Technology, Vellore, Tamil
Nadu, 632014

³ Department of Embedded Technology, Vellore Institute of Technology, Vellore
Tamil Nadu, 632014
chettiargautam@gmail.com

Abstract. With the recent advancements in medicine and biotechnology, there is always a growing need for more sophisticated decision systems to aid in certain biological factors, such as detecting a pathogen or microscopic organism and its accurate classification. In such a domain, there is very little tolerance for error, and the highest priority must be given to creating intelligent systems capable of doing so. Advancements in Computer Vision and other state-of-the-art Image Processing Techniques enable computation systems to extract and analyze principal features from a specimen that even a trained eye may otherwise miss. There has also been an upsurge in the research on the applications of Machine Learning to aid in specific surgical procedures and diagnostic tests. Classifying organisms at the micro-scale is an essential stepping stone for their long-term success. The study aims to ensembling the best Machine Learning techniques and models in the Computer Vision domain, such as Transfer Learning, top-performing models, and Image Preprocessing Methods, to maximize the capabilities of a typical Image Classification model for a sparse and limited multi-labeled class dataset containing microscopic images and hence shed light and explore into how more excellent performance can be obtained while facing classical dataset limitations.

Keywords: Image Preprocessing · Transfer Learning · Microorganism Dataset · Ensemble Models · TensorFlow · Computer Vision

1 Introduction

Microorganisms are ubiquitous. They are present all around us and are mostly imperceptible to the naked eye. They can be generally divided into two sub-categories, namely harmful and benign. The ones considered at the diagnostics level are generally the harmful ones, and timely and accurate detection can save

the patient from fatality. The benign ones are all around us, such as the food we consume, the water we drink, and so on. Yeast, probiotics, antibiotics, etc., are typical examples of such microorganisms in our daily lives. Advancements in computer vision prompt lesser dependence on the morphology of such microorganism data sources and more reliance on automated decision systems to discriminate such data.

This domain has had increasing amounts of research for a long time. The application of Multi-Layer Perceptrons (MLP) began this endeavor, and significantly more advanced Convolutional Neural Networks (CNN) architectures were introduced for the classification task.

The critical factor in the success of multi-class labeled datasets is the quantity of available data. For instance, the Environment Microorganism Image Dataset (EMDS-5) [1] consists of well-captured images of microorganisms, which contain original photos and the ground truth of those images. This allows significantly better performance due to the preprocessed segmentation feature that can be implemented using ground truth images. It contains 21 types of EM (Environment Microorganisms), with each class containing 20 kinds of original images per each EM and 20 GT (Ground Truth) Images.

However, with the sample space of such EM's increasing rapidly, today being at around 10^{12} specimens in the species, the need for a model which can adapt to more specific datasets without any significant structure or preprocessing and which can carry out classifications on such EM's which may have been captured from diverse sources is critical, and having sparse datasets combined to form another dataset based on labels alone simulates such a case very well. A very accurate representation of a dataset for EMs where multiple images within each class have significant variance yet belong to the same species is the Micro-Organism Image Classification Dataset [2] obtained from a Kaggle repository.

Similar datasets have been explored, and image classification tasks are carried out using optimal CNN architectures such as LeNet-5, VGG-16, ResNet-18, AlexNet, VGG-19, ResNet-34, and so on [3]. Another study also applied the computationally intensive DenseNet-201 model to classify EMs from the EMDS dataset while carrying out hyperparameter optimization for generating an optimally fine-tuned DenseNet-201 model called Optimally-Fine-Tuned DenseNet-201 (OFTD) [4].

Another recent study has implemented optimized deep-learning image models for a similar use case. The testing was conducted on computationally limited devices (Android phones) [5] and found significant performance due to model fine-tuning and pruning to handle the version and inferential time capabilities. The scope for research in this domain is still plentiful, and developments at any scale can significantly shape the future toward more intelligent systems that reduce morphology and ultimately depend on computing systems to nullify the scope for error. At the same time, the designs are practical for deployment at a civil level for general public use.

Multilayered Perceptrons (MLP) also have seen good results when coupled with higher-ordered features [6] for classification tasks for tabulated data. Trans-

formations for some 1-D and 2-D signals [7] may also be applied to generate better results depending on the nature of the data. However, generally, CNN models perform best [8] for image data, even for different types of biomedical image data. Some image processing techniques, such as CLAHE [9], can significantly boost the performance of even simple CNN models.

2 Literature Survey

Wang F. et al. [10] explore conjugate polymer-based approaches for pathogen sensing and detection based on structural variations at the peripheral level. This provides a significant advantage in the pathogenic discrimination process. The authors aim to extrapolate the approach in the discrimination of microscopic organisms in the detection and discrimination of pathogens such as the SARS-COVID-19 virus.

Ma P. et al. [11] carry out an elaborative survey of the detection techniques, which cover the breadth of image processing techniques and classical deep-learning approaches that have been worked upon in the domain of microorganism image analytics. They rightly note the complications of the detection task due to the cornucopia of such species (10^{11} - 10^{12}) and the acquisition complexity with accurate image representations from sources such as microscopes. Their survey encompasses techniques such as segmentation, image fusion, image transformation, and other critical methods in the detection process.

Kulwa F. et al. [12] propose a segmentation-based approach for classifying microscopic images christened PDLF-Net. The authors explore the RGB images following ground truth analysis and identify critical points used in patch generation. Transfer learning is then used via the application of VGG-16 on the generated patches, which is then used to create pairwise feature maps. Stacked over the base model is the SEGNet, which trains on the feature maps to develop segmented images of the ground truth. The imposed preprocessing techniques were applied to limited visibility microorganism images to produce higher-significance feature maps for the classification task.

Zhang J. et al. [13] propose an SEM-RCNN model for classifying multi-class microorganisms. They introduce novelty in the multi-class detection of EM by combining on top of the ResNet architecture a SENet block which instantiates a self-attention mechanism for optimal feature extraction. The approach includes a region proposal framework post-feature map selection. Transfer learning was implemented on the SENet model, which was pretrained. The authors finally used hyperparameter optimization to obtain suitable learning rates and epochs for maximum accuracy.

Prada P. et al. [14] review the recent advancements in methods of discrimination and detection of microorganisms. They explore a variety of bioprocesses as well as computation methods such as Raman Spectroscopy and the application of Deep CNN models in identifying complex microorganism samples. They also signify that fluorescent nanoparticles that attach themselves at localized target

particle sites and post-fluorescence-inducing processes make the classification task significantly more straightforward due to conspicuous lamination.

Shao R. et al. [15] propose a novel architecture based on compiling a vision transformer and CNN structure. They carry out EM (Environmental Microorganism) classification. Their proposed HTEM model is deployed in 4 stages, which begin with token embeddings, followed by a feature embedding layer, which is then fed into the transformer encoders and finally fed into a local FFN (Feed Forward Network) block. They outperform other CNN-based models and the Xception network by 9.02%.

Zhang J. et al. [16] explore applications of ANNs in recognizing microorganism images. Their review consists of the most popular Image Net models starting from MLP to CNN models and finally coming to the current vision transformers in the biological detection field, which aims to apply ML to supersede microorganism morphology. They explore publicly available datasets such as SIPPER, EMDS, and WHOI-Plankton.

Rani P. et al. [17] reviewed different automated methods for detecting microorganisms from image data. The authors also emphasize the performance of CLAHE and region extraction methods, such as the RF method, during the testing phase. They also generally support using Support Vector Machines with a linear kernel for the classification task. However, they recommend CNN-based models in the end.

Kulwa F. et al. [18] carried out a review of various segmentation techniques for image data for the detection of microorganisms. They employ different edge-based segmentation techniques such as Canny edge detection, etc. They also use binarization techniques such as Otsu Binarization for the ground truth analysis, which is later used in segmentation and helps provide proposed regions. They find promise in using models such as VGG-16 and U-Net.

Ponraj D. et al. [19] conducted a comprehensive survey of image preprocessing techniques for a biomedical image data use case. The authors explored the applicability of enhancement techniques such as CLAHE and operators such as Sobel, Gaussian, Laplacian, Prewitt, etc., for edge detection of the image data. They also explored image transforms such as the Gabor transform, wavelet transforms, curvelet transforms, etc.

3 Methodology

A model is proposed, which is an ensemble of optimally fine-tuned and pruned EfficientNetB7 via a Transfer Learning approach, the architecture of InceptionV3 with small amounts of pruning, and a custom Deep CNN architecture that is ensembled into one classical stacking structure which is topped off with a Voting Ensemble structure to take the best of three predictions which will act as the final decision of the whole ensemble. Even before the model description and creation, a significant amount of preprocessing is carried out on the Microorganism Dataset [2]. Transfer learning is highly successful in the classification task and preprocessing of the dataset images.

3.1 Dataset

The Microorganism Image Classification dataset consists of around 800 images of variable resolutions belonging to one of the eight labels: Amoeba, Yeast, Euglena, Spiral Bacteria, Hydra, Rod Bacteria, and Paramecium. There is an unequal distribution of data on each of the labels. The dataset used may be less impressive regarding the universe of discourse, the number of samples, or the quality of the pieces. The dataset images are of varying brightness, color saturation, noise levels, resolution, and intra-class correlative relevance. The requirement of pre-processing for obtaining image coherence for optimal model training is key to such data limited tasks. Its primitive nature and the high level of variance on a small subset of images make it challenging to train a model on it. Still, it is a good representation of how classifications would take place in real life, giving it significance for this purpose. The uneven class distribution is shown in Fig 1 and Fig 2.

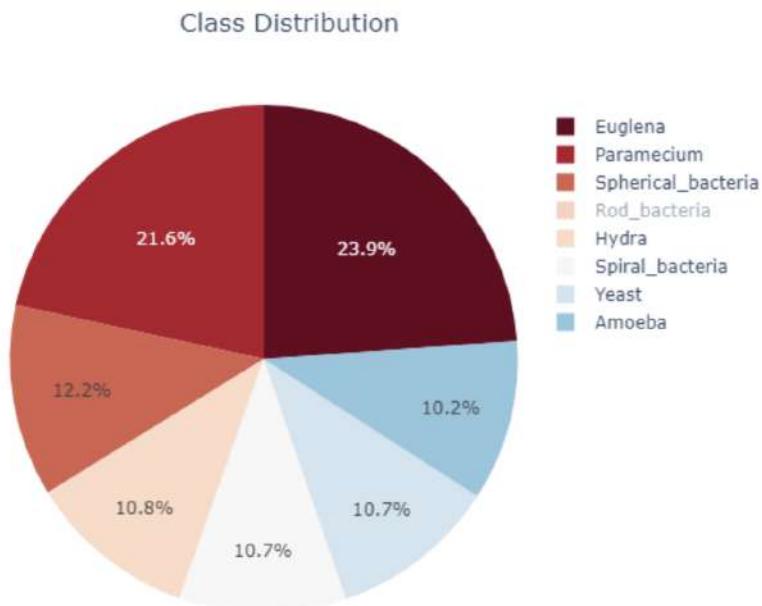


Fig. 1. Class distribution of the images in the dataset based on proportions.

As mentioned in the Introduction section, even images with the same labels have high variability and variance. It becomes a task to train a model under such data characteristics, especially given the small dataset size. An example training batch from the dataset is shown in Fig 3.

Fig 3. Sample batch from the training data. A model that can generate unique features for each label to distinguish the classes accurately while data limitation is a unique research aspect.

The sparseness of dataset can be seen by viewing multiple images within the same class. It is conspicuous that these RGB images, while under a single class, are weak representations of each other. Hence, sparsely limited data stems from the idea of having a small sample size that is not significantly intra-class correlated.

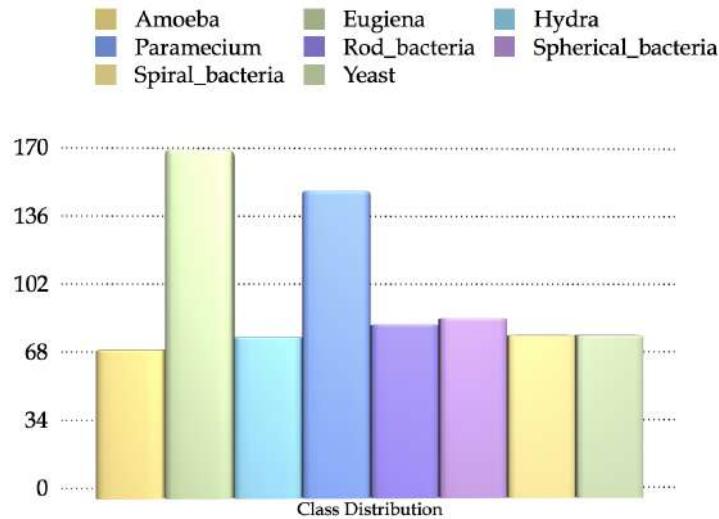


Fig. 2. Class distribution of the images in the dataset based on the count.

3.2 Data Prepossessing

Data Preprocessing The images in the dataset are limited in number. Hence, they must be appropriately preprocessed to utilize each of the microorganisms' components and generate significant feature maps during model training. For image preprocessing, multiple techniques are used.

First, the image is fed into a custom filter bank which consists of a Mean, Median, and Gaussian Filter. This proposed filter ensemble conjointly averages and eliminates the noise present in the 2D image data. Following this step, the image then undergoes saturation modulation to enhance the color vibrancy and sharpness by adaptively increasing the saturation of the image data. The saturation increment further intensifies the power of each of the color channels, and hence results in a brighter and more vivid colored image, hence increasing vibrancy. After this step, to tackle the lower resolution images, the highly efficient Super

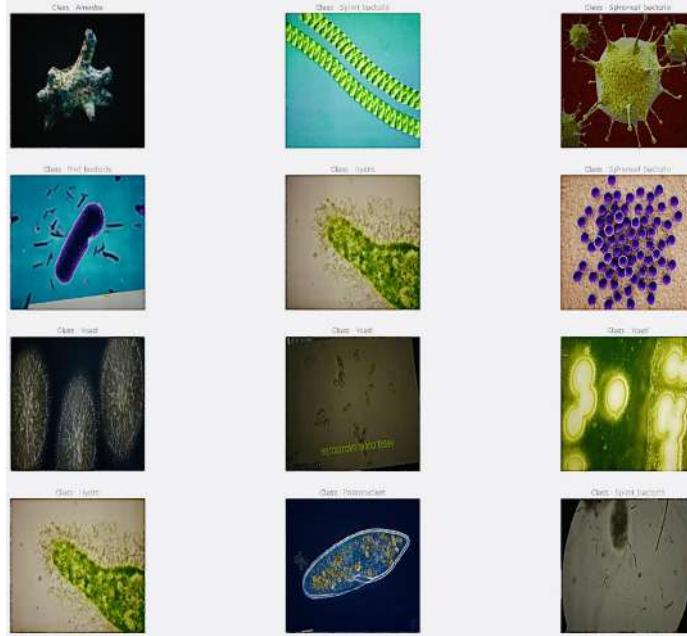


Fig. 3. Sample batch from the training data.

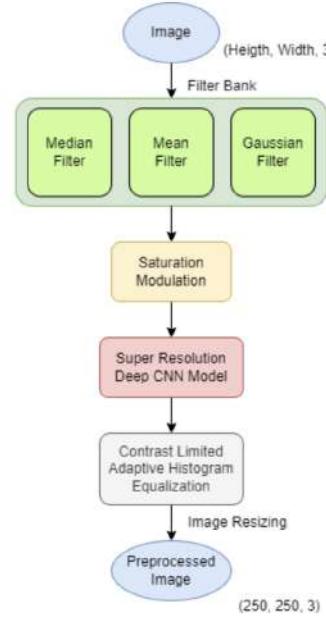
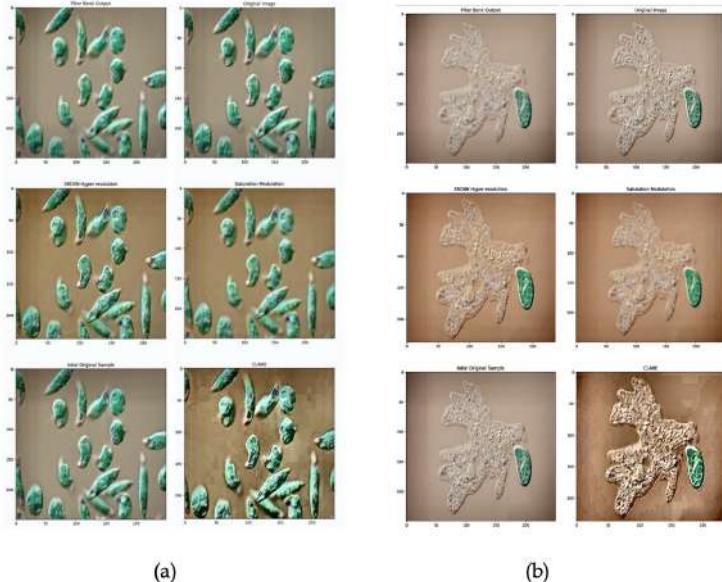
Resolution CNN (SR-CNN) model is used [20] with a transfer learning setup, and the resolutions of the images are significantly improved. The application of the SRCNN model is necessary for the dataset being used as some of the images present have lower resolution and quality and can provide more meaningful features during training if reconstructed properly. Finally, CLAHE is used to get sharper edges in the final stage of the picture. The entire preprocessing block is shown in Fig 4.

For filtering the unwanted noise in the image, the proposed approach uses a custom filtering ensemble. Some of the mathematical operations used are below. The equations for the Mean filter, Median Filter and Gaussian Filter are given in (1), (2,) and (3), respectively for each pixel $P_{i,j}$.

This preprocessing process flow is then tested against sample images to juxtapose their intermediate outputs at relevant stages to see the significance of the various blocks included in the process. The results of the preprocessing are shown in Fig 5 a and b.

From Fig 5 a and b, it's clear that the preprocessing block is generating significantly better images from the original rudimentary image data. These pre-processed images were then fed into model training and testing. This concludes the preprocessing carried out in the proposed approach.

All the class labels are converted into their One-Hot Encoded form to create a multi-class classification problem. This transforms the task into a probability

**Fig. 4.** Preprocessing setup for Image data**Fig. 5.** Filter Bank application, Saturation Modulation, SRCNN resolution modulation, and CLAHE output juxtaposed against the original image for (a) Euglena and (b) Ameoba

selection problem where the class with the highest probability is deemed the class obtained.

3.3 Model Creation

The approach aims at creating a model that can utilize the best of all types of learning. It can be seen that there are cases that two images within the same class are drastically different, but may not differ a lot from images from other classes. To tackle this classical problem, there is a need for a complex, robust, and feature sensitive image classification model. To overcome this, the proposed approach adopts a stacked ensemble structure and employs a Voting Classifier module as the final decision unit for the model. The model consists of three pipelines. The first is a transfer learning-based EfficientNetB7 model. The EfficientNetB7 consistently ranked well on the benchmarking tests on the MNIST dataset and is also computationally not too ambitious, making it a viable option after significant pruning to run on computationally weaker devices such as smartphones or other edge devices.

The second pipeline has an instantiation of an adapted version of the InceptionV3 architecture. The Inception model is better in some cases due to its smaller and factorized convolutions, which may take a little time for the training but require lesser computation resources for the model fitting and testing. The standalone InceptionV3 model also performs well on similar datasets as well.

Finally, the third pipeline is a distilled and pruned version of a VGG model, with much lesser layers and starts from smaller convolutions and builds up to medium size filters. It is computationally very inexpensive, has fewer features at the pooling stage, and produces significant overall performance results.

The EfficientNetB7 implemented architecture is shown in Fig 6a. The standard InceptionV3 architecture is used from the original description. The custom Deep CNN is shown in Fig 6b.

All the models use the Adam optimizer with the Sparse Categorical Cross-Entropy loss function during model compilation. Each of the standalone models is initially trained separately with 50 epochs and a batch size of 32. In the final ensemble, these models are then put in a Voting Classifier framework which uses hard voting for classification.

From Fig 5 a and b, it's clear that the preprocessing block is generating significantly better images from the original rudimentary image data. These pre-processed images were then fed into model training and testing. This concludes the preprocessing carried out in the proposed approach.

With all the base models and processing in place, the study proposed its complete ensembled model for the microorganism data fitting ad testing. The entire architecture is shown in Fig 7.

3.4 Results

The model was trained and tested on the image data from the dataset. Two hundred images were kept aside for testing purposes. Initially, the standalone

models were evaluated to test for performance, and finally, the ensemble model's accuracy was compared to individual components. The results found are shown in Table 1 below.

The results from Table 1 show the proposed ensemble's performance and viability. The reason behind the model's performance was the preprocessing techniques, the choice of supporting base models, and having a reliable voting classifier with the Softmax nature of predictions.

The final accuracy, loss, parameter and time quantum metrics of the individual and ensembled models can be seen from the box plots in Fig 8 a, b, c, and d.

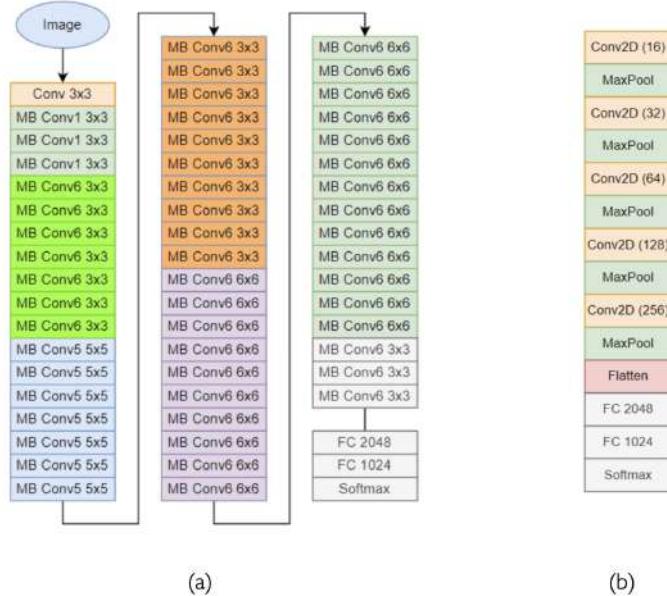
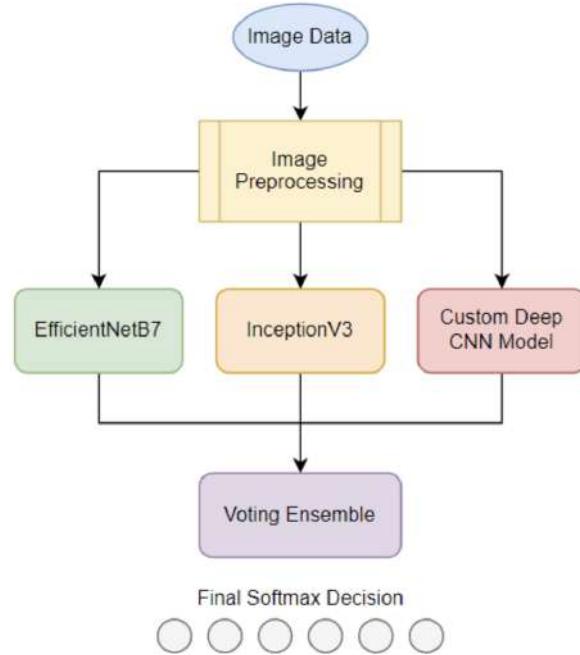
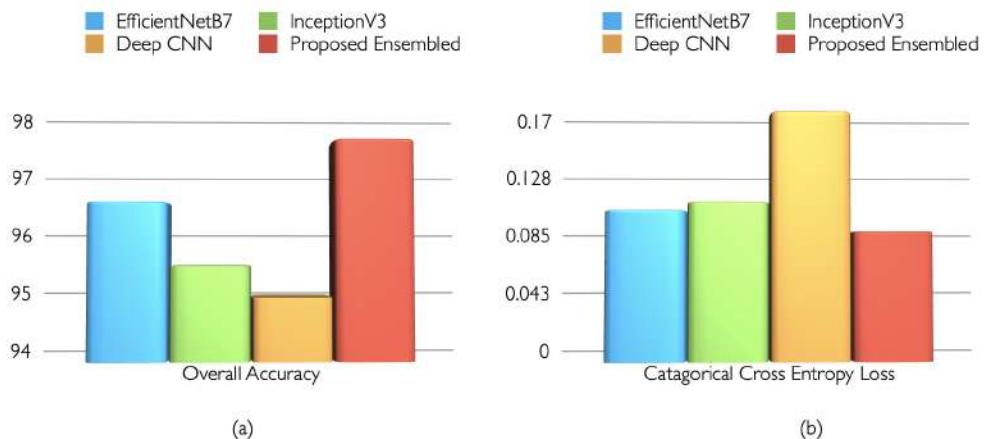


Fig. 6. Implemented architectures of (a) EfficientNetB7 and (b) Custom Deep CNN

Table 1. Performance metrics of the standalone and the proposed ensemble models.

Model	Parameters	Accuracy	Loss	Time per Step
EfficientNet B7	69 M	96.50%	0.1005	101 ms
InceptionV3	24 M	95.50%	0.1063	127 ms
Custom Deep CNN	5 M	95.00%	0.1670	14 ms
Proposed Ensemble	100 M	97.50%	0.0865	137 ms

**Fig. 7.** The proposed classification ensemble.

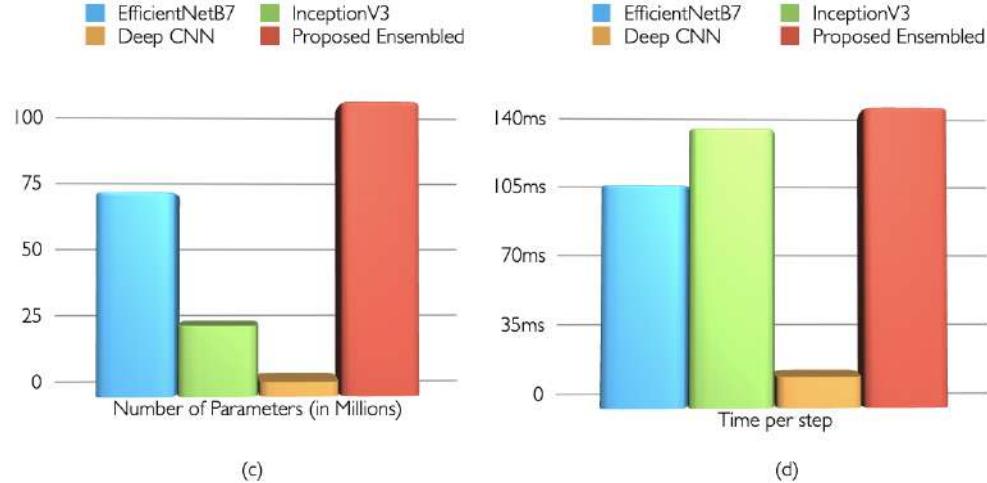


Fig. 8. Performance metrics in terms of (a) accuracy and (b) loss (c) Number of Parameters and (d) Time per step

4 Conclusion

There is a significant requirement for reliable and sustainable algorithms and methods in biomedical data, specifically the microorganism detection domain. The study only explores one of the many possibilities while considering a highly pragmatic dataset having high variability even for the data belonging to the same class. The proposed ensemble is not computationally too expensive, having just a little over 100 M parameters as shown in Fig 8c, and making predictions within a period of about 150 ms. It can also deliver significant results with minimal model training and a minimal data sample space and get up to 97.50% accuracy due to its inherent voting nature. The development of more advanced image preprocessing techniques based on either edge detection or region proposition or exploring more sophisticated models such as vision transformers poses higher accuracies and performance metrics for even sparser datasets having even more limited data. While it is traditionally believed that high-quality data with superior images is required to train a good-performance machine learning model, this approach shows that even sparse and limited data with the appropriate processing can obtain significant results. The goal of such research is to provide the medicine and biotechnology industry with pragmatic advancements and methods to aid in their morphology process, avoid any scope for human errors, and potentially save plenty of lives.

References

1. Li, Z., Li, C., Yao, Y., Zhang, J., Rahaman, MM., Xu, H., et al.: Environmental Microorganism image dataset Fifth Version for multiple image analysis tasks. PLoS ONE **16**(5), 99–110 (2021). <https://doi.org/10.1371/journal.pone.0254578>
2. Waquar, 2022, Micro-Organism Image Classification, Kaggle, <https://doi.org/10.34740/KAGGLE/DSV/4032122>
3. Poomrittigul, S., Chomkwan, W., Tanpatanan, T., Sakorntanant, S., and Treebupachatsakul, T., "A Comparison of Deep Learning CNN Architecture Models for Classifying Bacteria," 2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), 2022, pp. 290-293, <https://doi.org/10.1109/ITC-CSCC55581.2022.9894986>
4. Chen, W., Liu, P., Lai, C., Lin, Y., Identification of environmental microorganism using optimally fine-tuned convolutional neural network, Environmental Research, Volume 206, 2022, 112610, ISSN 0013-9351, <https://doi.org/10.1016/j.envres.2021.112610>
5. Kim, H.E., Maros, M.E., Siegel, F., Ganslandt, T. Rapid Convolutional Neural Networks for Gram-Stained Image Classification at Inference Time on Mobile Devices: Empirical Study from Transfer Learning to Optimization. Biomedicines 2022, 10, 2808. <https://doi.org/10.3390/biomedicines10112808>
6. Teja, K. U. V. R., Reddy, B. P. V., A, L. P., Patil, H. Y. and P. C. T, "Prediction of Diabetes at Early Stage with Supplementary Polynomial Features," 2021 Smart Technologies, Communication and Robotics (STCR), 2021, pp. 1-5, <https://doi.org/10.1109/STCR51658.2021.9588849>
7. Teja, K. U. Venkata Ravi., Reddy, B. P. Venkat., Alla, L. P. and Patil, H. Y., "Parkinson's Disease Classification using Quantile Transformation and RFE," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021, pp. 01-05, <https://doi.org/10.1109/ICCCNT51525.2021.9580024>
8. Hebbar, N., Patil, H. Y., and Agarwal, K., "Web Powered CT Scan Diagnosis for Brain Hemorrhage using Deep Learning," 2020 IEEE 4th Conference on Information & Communication Technology (CICT), 2020, pp. 1-5, <https://doi.org/10.1109/CICT51604.2020.9312098>
9. Patil, P. and Patil, H., "X-ray Imagining Based Pneumonia Classification using Deep Learning and Adaptive Clip Limit based CLAHE Algorithm," 2020 IEEE 4th Conference on Information & Communication Technology (CICT), 2020, pp. 1-4, <https://doi.org/10.1109/CICT51604.2020.9312089>
10. Wang, F., Ma, M., Cao, H., Chai, X., Huang, M., Liu, L., Conjugated polymer materials for detection and discrimination of pathogenic microorganisms: Guarantee of biosafety, Biosafety, and Health, Volume 4, Issue 2, 2022, Pages 79-86, ISSN 2590-0536, <https://doi.org/10.1016/j.bsheal.2022.03.006>
11. Ma, P., Li, C., Rahaman, M.M. et al. A state-of-the-art survey of object detection techniques in microorganism image analysis: from classical methods to deep learning approaches. Artif Intell Rev (2022). <https://doi.org/10.1007/s10462-022-10209-1>
12. Kulwa, F., Li, C., Grzegorzek, M., Rahaman, M., Shirahama, K., Kosov, S., Segmentation of weakly visible environmental microorganism images using pair-wise deep learning features, Biomedical Signal Processing and Control, Volume 79, Part 2, 2023, 104168, ISSN 1746-8094, <https://doi.org/10.1016/j.bspc.2022.104168>
13. Zhang, J., Ma, P., Jiang, T., Zhao, X., Tan, W., Zhang, J., Zou, S., Huang, X., Grzegorzek, M., Li, C. SEM-RCNN: A Squeeze-and-Excitation-Based Mask Region

- Convolutional Neural Network for Multi-Class Environmental Microorganism Detection. *Appl. Sci.* 2022, 12, 9902. <https://doi.org/10.3390/app12199902>
- 14. Prada, P., Brunel, B., Reffuveille, F., Gangloff, S.C. Technique Evolutions for Microorganism Detection in Complex Samples: A Review. *Appl. Sci.* 2022, 12, 5892. <https://doi.org/10.3390/app12125892>
 - 15. Shao, R., Bi, X-J., Chen, Z. (2022) A novel hybrid transformer-CNN architecture for environmental microorganism classification. *PLoS ONE* 17(11): e0277557. <https://doi.org/10.1371/journal.pone.0277557>
 - 16. Zhang, J., Li, C., Yin, Y. et al. Applications of artificial neural networks in microorganism image analysis: a comprehensive review from conventional multilayer perceptron to popular convolutional neural network and potential visual transformer. *Artif Intell Rev* (2022). <https://doi.org/10.1007/s10462-022-10192-7>
 - 17. Rani, P., Kotwal, S., Manhas, J. et al. Machine Learning and Deep Learning Based Computational Approaches in Automatic Microorganisms Image Recognition: Methodologies, Challenges, and Developments. *Arch Computat Methods Eng* 29, 1801–1837 (2022). <https://doi.org/10.1007/s11831-021-09639-x>
 - 18. Kulwa, F. et al., "A State-of-the-Art Survey for Microorganism Image Segmentation Methods and Future Potential," in *IEEE Access*, vol. 7, pp. 100243-100269, 2019, <https://doi.org/10.1109/ACCESS.2019.2930111>
 - 19. Ponraj, D. Narain, et al. "A survey on the preprocessing techniques of mammogram for the detection of breast cancer." *Journal of emerging trends in computing and information sciences* 2.12 (2011): 656-664.
 - 20. Chao, D., Loy, Change, C., Kaiming, H., Xiaoou, T. (2014). Image Super-Resolution Using Deep Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 38., <https://doi.org/10.1109/TPAMI.2015.2439281>

Assiduous Study of the Hyper-parameters' Influence on CNN using COVID-19 CT Images

Srinivasa L.Chakravarthy¹ [0000-0001-9141-4863], Varun Mallela¹, Vedula Sai Sarvanth¹, Rohith Sunkara¹ and Srimurari Dachepalli¹

¹ Dept. of CSE, GITAM University, Visakhapatnam, India
{chakri.ls, varunmallela08, sarvanthvedula, rohithsunkara06, murari.dachepalli} @gmail.com

Abstract. The SARS-CoV-2 virus causes the infectious COVID-19 disease. It is rapidly spread, and it resulted in a global pandemic. Every disease should be diagnosed so that the process of diagnosis can avoid long-term complications and also have a chance to increase the efficiency of the treatment. Similarly, COVID-19 should also be diagnosed and can be done in several ways. One of many ways is by Nucleic Acid Amplification Tests (NAAT). Due to some deficiencies in those tools or methods, many experts have recommended CT images as a diagnostic tool for detecting the COVID-19 virus more accurately and quickly. In this study, we used a dataset consisting of 349 COVID-19-affected CT scans and 463 regular CT scans. In classical computers, neural networks are powerful classification models. So, we have built a Convolution Neural Network model and achieved an accuracy of 98.6%. We have also tested the impact of various hyperparameters on the model, such as activation function, input shape, image rotation, number of layers, and epochs. We plotted different accuracy graphs for each parameter and found an optimal solution to be used in the model to improve the accuracy

Keywords: *COVID-19, Convolution Neural Network, CT scans, Hyperparameters, Classical computers.*

1 Introduction

COVID-19 is a dreadful disease that can cause mild to critical respiratory affliction and may lead to death; therefore, we should consider it seriously. Many people are now stating that we can go with our everyday lives as this virus minimizes, but some new variants are showing up. So, we decided to do our research on COVID-19 data [14]. For manual diagnosis of COVID-19, molecular tests need to be done. Molecular tests are also referred to as RT-PCR (reverse transcription polymerase chain reaction) test, NAAT (nucleic acid amplification) test, and RT-LAMP (reverse transcription loop-mediated isothermal amplification) test. These tests involve collecting samples from the nasal or throat, which some people are not comfortable with that, and even from collecting samples, one may have a chance to get an infection if there is no hygiene followed in the process. RT-PCR test is valuable and functional, but there are some chances of getting false results. Also, if there are excessive samples, it will take more time to get results. So, another alternative for diagnosis of COVID-19 is using CT scans

and X-rays [13]. CT-scan is also not wholly efficient and advisable as it causes X-ray radiation, but it can be used as an alternate method [20],[21]

Deep Learning is a subfield of Data Science. Deep learning incorporates predictive analysis modeling and statistics. It is a neural network that tries to simulate the human brain, and this network usually has a minimum of three layers or more. It still works for single or two layers, but by adding more layers, we get an optimal solution and accuracy [19]. In this field, the system learns from experience and trains its network [15]. At every stage, the amount of data keeps increasing, which increases system accuracy by training the model to more data. In deep learning, many types of neural networks exist, like convolutional, recurrent, and artificial networks. Our project mainly involves the recognition of CT images, classifying them, and predicting results by observing image patterns [16], [17]. All these tasks are significant functionality of a convolutional neural network. An essential feature of CNN is that it can discover image features and extract them without any manual help. In CNN, first, the simple patterns in the picture, like surroundings, edges, and lines, are identified, followed by complex crux patterns. CNN will be valid for this project by extracting patterns from COVID-19 and non-COVID-19 CT scans, and using this classification model will be trained to diagnose the virus.

Our objective is to build and train a model to identify the presence of COVID or not using CT scans of lungs and improve the model's performance by finding the best value for various input parameters. We considered layers, activation functions, epochs, input shape, and resize.

2 Related Work

Erdi ACAR and Ihsan YILMAZ [1] performed COVID-19 detection on various quantum real processors like IBMQx2, IBMQ-London, and IBMQ-Rome using the quantum transfer learning method. In this study, the ResNet18 convolutional network has been used as a feature extractor to obtain a final result of a 90% success rate in classical computers and 94-100% in quantum computers. In this study [3], the authors developed a deep learning-based automated classification model based on a convolutional neural network that exhibits a high COVID-19 detection rate. They used a data set that contains 3616 COVID-19 chest X-ray pictures and 10,192 images of healthy chest images. They have trained the data using various CNN models. Among all the deployed CNN models, MobileNetV2 produced the most remarkable accuracy of 98% in classifying COVID-19 and healthy chest X-rays.

In this study [4], chest X-Rays are used for detection instead of PCR tests. The dataset the authors have taken is inadequate to train deep neural networks. So, they used a concept called Domain Extension Transfer Learning (DETL). They employed DETL with a pre-trained deep convolution neural network on a large chest X-Ray dataset. In this paper [5], they have described pneumonia cases with chest X-rays or CT scans. The

dataset they used is used to study the progress of Covid-19 and how its radiological finding varies from other pneumonia such as MERS, SARS, and ARDS. They used the dataset to build tools that predict pneumonia and its outcome. The authors have conducted a study [6] to analyze the CT scans over time in Covid-affected patients. They have considered 90 patients with an average age of 45. A total of 366 CT scans have been acquired and studied by two radiologists for patterns, abnormalities, and CT scores. Their observation is that the lung abnormalities have peaked during the 6-11th days of illness.

To overcome the absence of professional doctors in distant regions and effectively identify COVID-19, Ozturk et al. [7] developed a novel method for the automated detection of COVID-19 using unprocessed chest radiography images. The authors of this work used the DarkNet model as their classifier and created 17 convolutional layers with various filters on each. The suggested model was created to offer precise diagnostics for binary and multi-class classification; the accuracies they got were 98.08% and 87.02%, respectively. The authors, Rahul Chauhan et al. [8], developed CNN models on image recognition and detection datasets to assess their performance. They used MNIST and CIFAR-10 datasets for recognition. The MNIST model's accuracy is 99.6%, and CIFAR-10's accuracy is 80.17%.

Authors Kang H, Xia L, et al. proposed to use a set of traits generated from CT scans to diagnose COVID-19 [9]. The suggested multi-view representation learning approach yielded 95.5%, 96.6%, and 93.2% in accuracy, sensitivity, and specificity, respectively. The author Sarra Guefrechi et al. [12] has utilized three robust networks, namely InceptionV3, VGG16, and ResNet50 have all been fine-tuned using compiled COVID-19 and standard casket X-ray pictures from several open databases. The experimental results show promise for the suggested models; they classified the X-ray pictures as Normal or COVID-19 with an accuracy of 97.20 for Resnet50, 98.10 for InceptionV3, and 98.30 for VGG16.

Ali Narin. et al [10] proposed Five pre-trained convolutional neural network-based models in this study (ResNet50, ResNet101, ResNet152, InceptionV3 and Inception-ResNetV2) for the detection of coronavirus pneumonia infected patients utilizing chest X-ray radiographs. According to the performance findings, the pre-trained ResNet50 model outperforms the other four models' classification performance. Ioannis D. Apostolopoulos et al. [11] examined the issue of automatically categorizing lung disorders from X-ray scans, including the recently discovered COVID-19. They used Mobile Net to imply that training CNNs from scratch may disclose important biomarkers connected to the COVID-19 disease but not exclusively. The overall classification accuracy of the seven classes is 87.66%.

3 Methodology

In this study, we train and validate a CNN model to classify CT images into COVID-19 and non-COVID-19 [18],[21]. Figure 1 provides an overview of the method of this study. To build a CNN, first, we added convolutional layers to work with a 2D image. Next, we added a pooling layer to reduce the size of the feature map by 2. We have taken the minimum size of the pool to retain the information in the image. Later we added several convolutions and pooling layers to improve the model's accuracy. We have added a flattening step to get the input node of connected layers. We also added a hidden and drop-out layer to prevent overfitting and implemented the output layer. Figure 1 illustrates an implementation of the CNN model for the binary classification of COVID-19 [22].

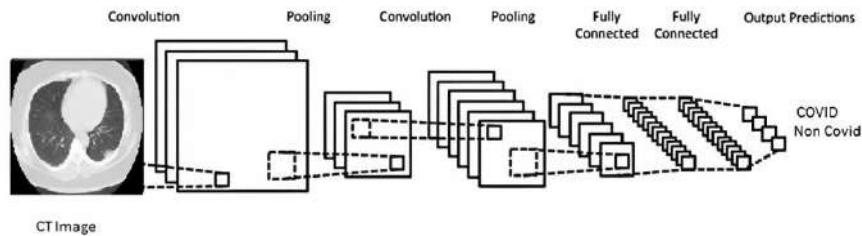


Fig. 1. CNN Model Implementation for classification of COVID-19 and non-COVID-19.

3.1 Data Set

We have used an open-source dataset named 'COVID-CT' [2] to train and validate the model. This dataset contains 349 CT images of COVID positive from 216 patients and 463 non-COVID-19 CT images. A senior radiologist has confirmed the utility of this dataset in Tongji Hospital, Wuhan, China, and has diagnosed and treated many COVID-19 patients during the outbreak of this disease between January and April. We have also referred to various other COVID CT image datasets [23]. Figure 2 illustrates two examples of CT images taken from the dataset.

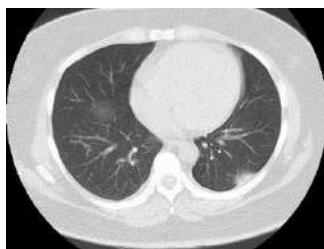


Fig. 2. (A) COVID-19 CT image. **(B)** Non-CT image

3.2 Input Parameters

We have tested various values for different parameter inputs to improve the model's training and validation accuracy [25]. We tried multiple parameters like the number of layers, epochs, input shape, and image orientation. The goal was to observe the model's behavior with changes in the values of different hyperparameters.

3.2.1. Layers and Epochs

A Neural network is a collection of input and output units that are connected, and in which each node has a weight. Each neural network model comprises one input layer, one output layer, and several hidden layers. The inputs go through the input layer before being simultaneously weighted and fed to the hidden layers. The inputs to the output layer are the weighted outputs of the final hidden layer.

Here, we have implemented several neural network models with a different number of layers in each model. At first, we implemented the neural network model with only one layer (no hidden layers), then we added a hidden layer. We continued this approach till we got similar values. We observed that as the number of layers increases, the time taken to run the model decreases. The number of layers is inversely proportional to the time taken or the model's speed.

We have also considered the epochs. The term epoch is the number of passes the machine learning algorithm has made across the training dataset. Here, an epoch is how often the algorithm came across each image. Instead of taking a certain number of epochs, we have kept the model in the for loop where the epoch gets incremented by 10 in each iteration.

The methodology adopted is being implemented in both aspects together. At first, we considered the neural network model with only one layer. We have fit the model in the for loop where the number of epochs changes per iteration. In the first iteration, we have considered ten epochs; in the 2nd iteration, we have considered 20 epochs, and this process continues till the 50 epochs (5 iterations). The exact process is implemented on the models by increasing the layers

3.2.2. Activation Function

Another parameter that we analyzed is activation functions. We employed three activation functions for the input layer, whereas we used two for the output layer. In total, we got 6 cases only for this parameter. We tested for all six cases at the same values for batch_size, steps_per_epoch, and the same number of epochs. Our objective for this parameter is to obtain an optimal activation function. We can get an optimal solution when accuracy is high, and there is no significant overfitting. We have calculated the training and validation accuracy of various activation functions for 20 epochs with 50 steps_per_epochs and ten validation_steps.

The most common functions used are Relu for the input layer and sigmoid for the output layer. We have observed better accuracy (training and validation) for this case than in other cases. Even for the Tanh function, we got high accuracy but some overfitting. We got low accuracy for the sigmoid activation function (for the input layer), i.e., fifty percent. We got the same accuracy for all three input functions when we used the softmax function for the output layer. Compared to other scenarios, the time required to train the model using the Relu activation function for 20 epochs is long.

Relu is the most used activation function as it is simple and fast. It returns 0 for negative value input and returns the same value when it receives a positive value. Simply we can write it as $f(x)=\max(0,x)$. The sigmoid activation function is also known as the logistic function. Input for this function can be any real value, and output values will be 0 to 1. It is used for both input and output layers. Tanh function is also called a hyperbolic tangent function. It is the same as the sigmoid function of input values, but this output will be from -1 to 1.

3.2.3. Input Shape

Input Shape is the dimension of the input image given as input to the Convolutional Neural Network. This image data is processed in various layers to detect important features of the image. The input dimension contains three parameters – width, height, and the number of channels. Here we have used channel three and tested for all images. We have tested different values of input shapes like (100,100,3), (200,200,3), (250,250,3), (300,300,3), (400,400,3), (500,500,3). For all the results, we have used 20 epochs with every 32 steps.

3.2.4. Image Orientation

The image orientation is a value representing the degree of rotation of the image. We have experimented with different degrees, 0,45,90,135,180. The training accuracy and validation accuracy are different for other degrees. They sometimes increase and sometimes decrease, and time varies with an increase in degrees. We have incorporated horizontal flips to flip images horizontally randomly. We have trained the method using 30 epochs, each consisting of 50 steps.

4 Results

We have divided the dataset into three groups. We have allocated 500 (71%) images for training, 100 (15%) images for validation, and 96 (14%) images for testing. We initially used a CNN model with four convolutional and four max pool layers to get accurate results. The activation function used was Relu, and the input shape was (150,150,3). We have used 20 epochs and 32 steps per epoch. Using these inputs, we obtained a training accuracy of 98.6% and a validation accuracy of 69%. The plot of training and validation accuracies of the model can be viewed in fig 3.

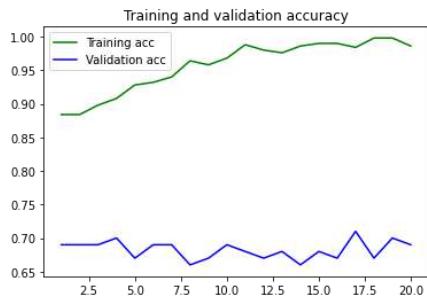


Fig. 3 The plot of training and validation accuracy of the model

4.1 Effect on the model for change in Epochs and Layers

As said earlier, we have considered the model with one layer and then created different models by incrementing the layers. In each model, we have considered 150 epochs; instead of taking all 150 epochs in a single iteration, we have taken ten epochs in the first iteration, 20 in the second, and so on to 50, which took five iterations in total. We have considered it in five iterations because we can calculate the time taken by the model for each iteration. As the layers increases, the time taken will be decreased; that is, the speed of the model increases. It took nearly 2 hours and 40 minutes to run a model with one layer, and time gets better as the number of layers increases. It took only 1 hour and 20 minutes to run all 150 epochs when we used a 5-layer model, and as the number of epochs increases, the model's accuracy improves. Our observations revealed that five layers and 50 epochs had yielded optimal results. Table 1 below presents all the accuracies for various layers and epochs. Figure 4 illustrates the training and validation accuracies for an optimal number of layers and epochs.

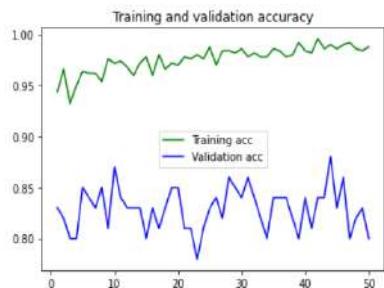


Fig. 4 The plot of training and validation accuracy for five layers and 50 epochs.

Table 1. Accuracies for Various values of the number of layers and epoch

Layers	Epochs	Training ACC	Validation ACC	Time Taken (sec)
1	10	0.7960	0.7300	0525.178
1	20	0.9960	0.7300	1307.459
1	30	0.9900	0.7300	1865.404
1	40	0.9920	0.7300	2752.529
1	50	0.9920	0.7400	3104.194
2	10	0.8380	0.7000	0377.077
2	20	0.9600	0.7500	0754.744
2	30	0.9680	0.7500	01122.74
2	40	0.9800	0.7500	1524.468
2	50	0.990	0.7600	1919.499
3	10	0.7340	0.7100	0324.433
3	20	0.8660	0.7800	0676.332
3	30	0.9300	0.7500	1057.160
3	40	0.9660	0.7500	1449.042
3	50	0.9700	0.8400	1714.577
4	10	0.5940	0.5500	0330.513
4	20	0.7220	0.7500	0704.983
4	30	0.8220	0.8100	0918.060
4	40	0.8960	0.8100	1327.355
4	50	0.9620	0.8000	1620.744
5	10	0.4640	0.5300	0326.641
5	20	0.6180	0.6500	0684.216
5	30	0.7780	0.7800	0909.295
5	40	0.9400	0.8400	1300.090
5	50	0.9880	0.8000	1525.510

4.2 Effect on the model for changes in the activation function

Relu has shown to be more accurate than sigmoid and Tanh in training and validation. We obtained good accuracy even for the Tanh function, albeit with considerable overfitting. We got 50% accuracy for the sigmoid activation function (for the input layer). When we applied the SoftMax function for the output layer, we acquired an accuracy of about 50% for all three input functions. Compared to other examples, the time needed to train the model using the Relu activation function for 20 epochs is high at 753 seconds. Therefore, Relu and Sigmoid are the best activation functions for the input and output layers, respectively. Figure 5 illustrates the training and validation accuracies for the optimal input and output activation functions. Table 2 below presents all the accuracies for different activation functions.

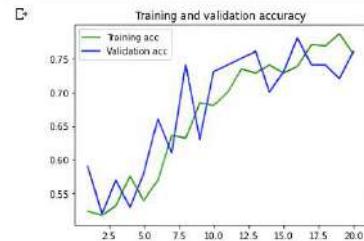


Fig 5. Plots of training and validation accuracy when input and output activation function are Relu and Sigmoid

Table 2. Accuracies for Various Activation Functions

SNO	I/P Function	O/P Function	Training ACC	Validation ACC	Time Taken (sec)
1	Relu	Sigmoid	0.7560	0.7600	753.24
2	Relu	SoftMax	0.5000	0.5100	693.75
3	Sigmoid	Sigmoid	0.5020	0.4900	699.72
4	Sigmoid	SoftMax	0.5000	0.5100	716.41
5	Tanh	Sigmoid	0.8340	0.7200	706.30
6	Tanh	SoftMax	0.5000	0.4800	697.02

4.3 Effect on the model for change in input shape

To check the effect of various values of input size on the model, we have used 20 epochs for each value and 32 step sizes. It was observed that training accuracy and execution time increased as the dimension of the input shape increased. Maximum accuracy was seen when the input shape was (500,500,3), taking around 4900 seconds for execution. It was also observed that the accuracy values remained the same beyond the input (500,500,3). The optimal input shape found was (250,250,3). This input shape produced an accuracy of 98.6% taking around 1597 seconds, as it has a high accuracy enough for the model validity. Figure 6 illustrates the training and validation accuracies for optimal dimension of the input shape. Table 3 below presents all the accuracies for various values of inputs values of input size.

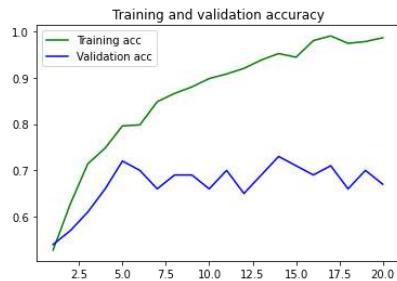


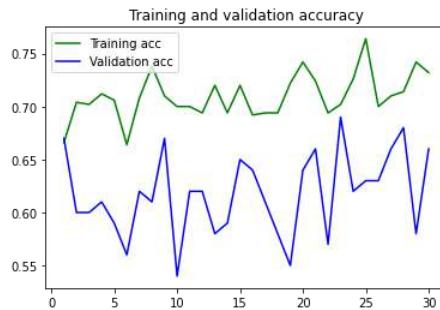
Fig 6. Plots of training and validation accuracy when input shape is (250,250,3)

Table 3. Accuracies for Various values of Input Size

SNO	Input Shape	Training ACC	Validation ACC	Time Taken (Secs)
1	(100,100,3)	0.9020	0.6700	325
2	(200,200,3)	0.9840	0.6900	1529
3	(250,250,3)	0.9860	0.6700	1597
4	(300,300,3)	0.9900	0.6500	2767
5	(400,400,3)	0.9840	0.7700	3768
6	(500,500,3)	1.0000	0.7900	4893

4.4 Effect on the model for change in image orientation

These are the Results for the parameter's different rotation ranges. We considered the model with four layers and then created other models with varying rotation ranges. In each model, we have considered 30 epochs. We have considered it in five iterations to calculate the time taken by the model for each iteration. We have found that the accuracy continuously increases with an increase in rotation range. As the rotation range increases, the time taken will also decrease; that is, the model's speed increases. The model's execution took maximum time, i.e., 17 minutes, with a rotation range of 180° . Through our research, we found that 135° rotation of the image has yielded the optimal results. Figure 7 illustrates the training and validation accuracies for the optimal CT image orientation. Table 4 below presents all the accuracies for various values of image orientation.

**Fig 7.** Plots of training and validation accuracy when image orientation is 135° .**Table 4.** Accuracies for various values of Image Orientations

SNO	Angle of Rotation	Training ACC	Validation ACC	Time Taken (Sec)
1	0°	0.6300	0.5800	1093
2	45°	0.6400	0.6600	1024
3	90°	0.7000	0.6500	995
4	135°	0.7300	0.6600	1017
5	180°	0.7400	0.6400	1080

5 Conclusion

The accurate testing of COVID-19 can be done through CT scans. In this paper, we built a CNN model for diagnosing COVID-19 using CT images to be run in a classical machine. We have taken the data required for modeling from an open-source COVID-CT dataset. Initially, we made and trained a CNN model to obtain a training accuracy of 98.6% and a validation accuracy of 69%. This obtained accuracy shows that our model can accurately diagnose COVID-19 and is suitable for clinical purposes. We have continued our research by observing the impact of various hyperparameter values like activation function, input shape, image orientation, number of epochs, and layers on the model. We have found optimal solutions for each parameter to improve the model's accuracy. According to our observations, we found that using five input layers, 50 epochs, Relu as input layer activation function, and sigmoid as output layer function, (250,250,3) as the input shape and 135° rotation of the CT image have yielded the best optimal results. In the future, we wish to implement and run this model in GPUs and Quantum Machines and compare its performance to classical computers.

References

1. ACAR, ERDİ and YILMAZ, İHSAN (2021) "COVID-19 detection on IBM quantum computer with classical-quantum transferlearning," *Turkish Journal of Electrical Engineering and Computer Sciences*: Vol. 29: No. 1, Article 4.
2. <https://github.com/UCSD-AI4H/COVID-CT>
3. Akter, Shamima, F. M. Javed Mehedi Shamrat, Sovon Chakraborty, Asif Karim, and Sami Azam. 2021. "COVID-19 Detection Using Deep Learning Algorithm on Chest X-ray Images" *Biology* 10, no. 11: 1174. <https://doi.org/10.3390/biology10111174>
4. Sanhita Basu, Sushmita Mitra, and Nilanjan Saha "Deep Learning for Screening COVID-19 using Chest X-Ray Images", 2020
5. J. P. Cohen, P. Morrison, and L. Dao, "COVID-19 image data collection," arXiv 2003.11597, 2020.
6. Y. Wang, C. Dong, Y. Hu, C. Li, Q. Ren, X. Zhang, H. Shi, and M. Zhou, "Temporal changes of CT findings in 90 patients with COVID-19 pneumonia: A longitudinal study," Radiology, p. 200843, 2020. PMID: 32191587.
7. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR (2020) "Automated detection of COVID-19 cases using deep neural networks with X-ray images." *Comput Biol Med* 121:103792
8. Rahul C, Kamal Kumar G, Joshi R.C (2018) "Convolutional Neural Network (CNN) for Image Detection and Recognition." DOI: 10.1109/ICSCCC.2018.8703316
9. Kang H, Xia L, Yan F, Wan Z, Shi F et al. (2020) "Diagnosis of Coronavirus Disease 2019 (COVID-19) With Structured Latent Multi-View Representation Learning. IEEE Trans Med Imaging 2020"; 39(8):2606-2614. doi:10.1109/TMI.2020.2992546
10. Narin, A., Kaya, C. & Pamuk, Z. Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Anal Applic* **24**, 1207–1220 (2021). <https://doi.org/10.1007/s10044-021-00984-y>

11. Apostolopoulos, Ioannis D., Dimitris J. Apostolopoulos, and Nikolaos D. Papathanasiou. 2022. "Deep Learning Methods to Reveal Important X-ray Features in COVID-19 Detection: Investigation of Explainability and Feature Reproducibility" *Reports* 5, no. 2: 20. <https://doi.org/10.3390/reports5020020>
12. Guefrechi S, Jabra MB, Ammar A, Koubaa A, Hamam H. Deep learning based detection of COVID-19 from chest X-ray images. *Multimed Tools Appl.* 2021;80(21-23):31803-31820. doi: 10.1007/s11042-021-11192-5. Epub 2021 Jul 19. PMID: 34305440; PMCID: PMC8286881.
13. Ai T et al (2020) Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases. *Radiology*: 200642
14. A. Bernheim, X. Mei, M. Huang, Y. Yang, Z. Fayad, N. Zhang, K. Diao, B. Lin, X. Zhu, K. Li, et al., "Chest ct findings in coronavirus disease-19 (covid-19): relationship to duration of infection," *Radiology*, p. 200463, 2020.
15. O. Gozes, M. Frid-Adar, H. Greenspan, P. D. Browning, H. Zhang, W. Ji, A. Bernheim, and E. Siegel, "Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning ct image analysis," *arXiv preprint arxiv:2003.05037*, 2020
16. X. Xu, X. Jiang, C. Ma, P. Du, X. Li, S. Lv, L. Yu, Y. Chen, J. Su, G. Lang, et al., "Deep learning system to screen coronavirus disease 2019 pneumonia," *arXiv preprint arXiv:2002.09334*, 2020.
17. M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. Al-Emadi, et al., "Can ai help in screening viral and covid-19 pneumonia?," *arXiv preprint arxiv:2003.13145*, 2020.
18. O. Gozes, M. Frid-Adar, H. Greenspan, P. D. Browning, H. Zhang, W. Ji, A. Bernheim, and E. Siegel, "Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning ct image analysis," *arXiv preprint arxiv:2003.05037*, 2020
19. J. Chen, L. Wu, J. Zhang, L. Zhang, D. Gong, Y. Zhao, S. Hu, Y. Wang, X. Hu, B. Zheng, et al., "Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography: a prospective study," *medRxiv*, 2020.
20. F. Shan+, Y. Gao+, J. Wang, W. Shi, N. Shi, M. Han, Z. Xue, D. Shen, and Y. Shi, "Lung infection quantification of covid-19 in ct images with deep learning," *arXiv preprint arxiv:2003.04655*, 2020.
21. S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng, et al., "A deep learning algorithm using ct images to screen for corona virus disease (covid-19)," *medRxiv*, 2020.
22. Hamza, Hentabli & Salim, Naomie & Nasser, Maged & Saeed, Faisal. (2020). Meramalnet: A Deep Learning Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery. 21-37. 10.5121/csit.2020.100203.
23. J. P. Cohen, P. Morrison, and L. Dao, "Covid-19 image data collection," *arXiv preprint arxiv:2003.11597*, 2020.

Academic Dishonesty Detection in Exams using Pose Extraction

*Dhanush Binu¹, Dr. Sivaiah Bellamkonda¹
¹ Department of Computer Science & Engineering
Indian Institute of Information Technology Kottayam
Kerala, India
^{1*} dhanush2019@iiitkottayam.ac.in
²sivaiah.bk@gmail.com

* Corresponding Author

Abstract. Academic dishonesty has been a significant setback to the proper functioning of educational institutions worldwide, as cheating in examinations is a significant and persistent issue contributing to academic dishonesty. With cases of cheating in examinations increasing year after year, it can be asserted that traditional invigilation and monitoring methods are ineffective. This work proposes a method to analyze cheating behavior in examination halls through pose estimation. First, we analyze the frame for any movement using a motion detection algorithm. The marked frame is sent to the pose extraction phase, where the pose of the person/persons present in the frame is extracted and given to the machine learning model, which identifies whether a student has been cheating. Furthermore, the frame marked with face recognition is stored for later use, along with the time and name of the student, in a separate file for reference.

Keywords: Pose Extraction, Computer Vision, Inter-frame difference method, Machine learning.

1 Introduction

Every year, thousands of examinations are conducted worldwide, ranging from small class tests to more prominent ones such as civil services examinations, entrance examinations, and many more. More than 60.8% admitted to cheating on tests, but only 95% of students are caught, and most students engaging in these acts have high grades. This issue could evolve into a significant issue as it could cause a decrease in the grades of motivated students, causing them to have lower percentile scores. Teachers have a hard time trying to catch those deceptive students. Hence, in most cases, they are relieved with little to no consequences due to professors/teachers having little to no evidence. In order to counter this, video-based surveillance has been proposed. Christopher et. al. [1] try to answer several

questions about video-based surveillance systems' efficacy. It was concluded that video surveillance is helpful since it provides investigative evidence, dissuading students from copying since they are being watched. Making efficient use of this conventional technology method of video surveillance is not advised since it is also prone to human errors. The paper concluded that the efficiency of automated Closed Circuit Television (CCTV) systems would be significantly improved to conventional usage.

Before the emergence of deep learning techniques, most research papers used IoT methods for cheating detection. Jiabul Hoque et. al. [2] use a high-sensitive microphone and 360-degree CCTV cameras to monitor and control the examinations. It also makes use of biometric readers for authentication purposes. With the proposed system, the number of invigilators can be severely reduced, and the examination can be controlled more conveniently. Fang et. al. [3] use a monitoring and seat calibration module with an alarm that goes off if any suspicious activity is detected. This article proposes an improved model that can extract and analyze the scene of unusual information features and provide a threshold algorithm to fine-tune the alarm to detect any abnormalities in the examination hall.

The development of Computer Vision has led to intelligent video-surveillance systems [4,5]. Intelligent video surveillance has been in commercial use for a long time. Nevertheless, intelligent video surveillance in classrooms has yet to be implemented, even though researchers are doing relevant research.

Exam cheating mainly involves students taking the help of some external source to solve exam questions. The techniques for cheating differ between an online and an offline(on-site) exam, but the most common cheating techniques are writing on hand or referring to small pieces of paper. Students tend to hide these papers in their pockets. After they are done referring to these papers, they discard them, or in the case of hand-writings, they rub them off. Most prefer to look into their other paper by turning their head, trading papers that contain answers and communication through gestures since these are much easier and have comparatively low risk as opposed to writing on pieces of paper. Douglas et. al. [6] analyze all the different techniques by which a student tries to cheat. The most prominent methods detailed in this paper are through the use of (i) body, (ii) seating arrangement, (iii) by use of foreign materials, and (iv) by use of technological devices. Since, in most cases, the student actively moves some parts of the body, analyzing one's posture can yield valuable insights into how a student copies. Human Pose estimation is a field in computer vision that has seen tremendous amounts of research in recent years. Cao et. al. [7] proposed a human pose estimation system to identify and track the key points of a person. These key points can be stored and used to monitor changes in a person's body. Human Pose Estimation has been used in many applications such as Fall Detection, Activity or

fitness monitoring systems. Research in human pose has reached a point where libraries such as TensorFlow and P5 are used to implement pose estimation and gait analysis in browsers, as described in [8]. This paper investigates different ways to predict and analyze different methods to detect cheating and proposes a novel method using human pose detection.

2 Related Work

Samir et. al. [9] implement a multi-person proctoring system. The model first takes input from the webcam. The input may consist of single or multiple people. The model then decides whether to single or multiple pose estimation. Pose extraction is done using the Tensorflow-PoseNet model on the input footage. The key points extracted are analyzed using a complex algorithm that considers the essential keypoints: the nose, eyes, and ears. The algorithm then applies equidistance formulas to these key points to calculate abnormal positions. The algorithm applies the equi-distance formula since the key points scale with distance off-camera. The model also tracks hand position using the x-axis and y-axis position of the wrist keypoints. If the x-axis of the wrist keypoints is far from the head, and the y-axis is not exceeding the head level, then abnormality is reported. The model was tested in varying environments and with different cheating variables that can be controlled. The model performed well in both experiments, with an average detection accuracy of 95% and 92.3% in abnormal head and hand movement.

Maniar et. al. [10] combine various models, such as Eye Gaze tracking, Mouth movement detection, Head Pose Estimation, Object detection, and Person counting model, to detect if a student has engaged in malpractice. Once the student is flagged twice, he/she will be warned. If the student continues to get flagged, the exam will stop, and she will not be able to continue writing the exam. The System will send the user data to the cloud so the teacher can cross-check the candidate. Sometimes, the user may be flagged due to technical issues and false model predictions. In that case, invigilators can manually check each candidate. Face detection is the key feature since it depends on eye tracking and mouth movement detection. Hence several models were compared, and the DNN module in OpenCV provided the best result. For Eye Gaze detection Dlib is utilized since it can make predictions in real-time. As far as Mouth movement detection is concerned, the distance between the key points of the lips is recorded and measured.

For object detection, pre-trained yolov3 is used. For Head pose detection, the angle of the head is recorded. There is also an audio-to-text converter that uses Google's speech recognition API to record audio and convert it into text. Words

spoken by the examinee are compared with the question paper, and the common words are reported to the invigilator. Nishchal et. al. [11] propose a model that consists of Posture detection using OpenPose, ALEXNET model for detecting the type of cheating, Emotion analysis, face recognition, and finally, report generation. A custom dataset was created with over 1000 normalized images of size 224*224. There are four main types of classes (Bending back, stretching the arms behind, Bending down, and Facing the camera). Kohli et. al. [12] implement a 3D Convolutional neural network to classify abnormal gestures and objects from a training sample of size 7638. The paper also draws a comparison between LSTM and RNN networks. Riaz et. al. [13] apply a pre-trained model for pose detection and key point extraction. These key points are then passed on to a high-dimension convolutional neural network. Mahmood et. al. [14] implement a cheating detection algorithm using a regional convolutional neural network. This neural network detects if the examinees are cheating using head movements and are mainly used for detecting the students in the examination hall. The paper also implements a face detection and recognition system using Multi-Task Cascaded Neural Network. The model is trained exclusively to monitor and control many students in a single frame. With the onset of the pandemic, research on online exam proctors has accelerated. Razan et. al. [15] implement an online exam supervision system that uses eye-tracking software to calculate the student's gaze. The student has cheated if any abnormality is detected in the values obtained from the eye-tracker.

3 Proposed Work

3.1 Dataset

The surveillance video of the examination room is typically unavailable to the public due to reasons concerning applicants' privacy. As a result, the related data cannot be collected in publicly available datasets. Hence, a custom dataset is prepared and used in this work, consisting of images taken with different angles and lighting with an equal number of images for each of the five labels. Image augmentation has been done on the dataset using the augmentor library to increase the training samples so that the model can diversify between minute details. The dataset was created with images of a single person for more accurate key point extraction. Every image is then scaled to a height of 800 with the aspect ratio preserved.

3.2 System Overview

The Input from the Video Surveillance Camera is passed through a motion detector, which captures frame samples of where the movement has occurred. This frame is passed through a face detection algorithm, which helps identify the student. The frame is then passed through a Pose extraction pipeline which outputs the key points.

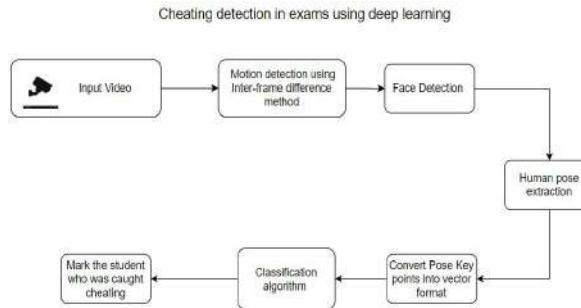


Fig. 1. Working pipeline of the system.

These key points are then analyzed to determine if the student has cheated, then the student's name and the time are noted on an excel sheet. The following sections explain each of the above models in detail.

3.3 Inter-frame difference method

Inter-frame difference method is used to detect any motion that has happened in the video input. In practice, the current and previous frames is taken and the `absdiff` method in OpenCV is used to find the difference and then dilute it. The output obtained is then passed through a threshold function, where the minimum and maximum threshold values. A threshold value can be adjusted so that small amounts of unnecessary motion get filtered out. Finally, the algorithm checks for any contours which could suggest if any motion was captured on the frame, and if the amount of contours in the image is greater than one, then the image is passed on to the Pose Extraction Stage. Nakashima et. al. [16] describe how to detect motion using the inter-frame difference method. They first determine the difference between the present and previous frames and binarize the resultant. The binarized image is then expanded and contracted for better processing. Using the careful observation of the histogram, the model can determine if any motion was present in the frame and also calculate the object's position in motion.

3.4 Face Recognition

Face Recognition was implemented using the face-recognition module in python. This module provides functions to retrieve the face encodings and the location of each face in the image. This encoding is then compared with the encoding of the faces stored in the project directory, and if it matches, then the model stores this information. It is easy to use and provides 99.38% accuracy on the Labeled Faces in the Wild Dataset. With this module, the tolerance/ sensitivity. Real-time face recognition can also be performed using this module. The snapshot of the person found cheating is captured and stored in a separate directory. The model takes these images and compares them in real time. If the person matches, the frame is stored in another directory for later inspection. The person's name and the occurrence's time are recorded and stored as a comma-separated value (CSV) file.



Fig. 2. Working demonstration of face recognition system

3.5 Pose Extraction

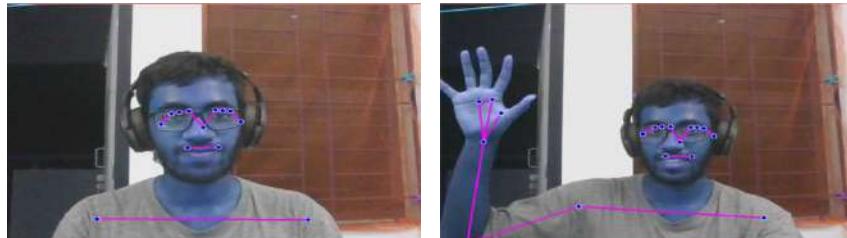


Fig. 3. Demonstration of Pose Extraction model

After several experiments, it was found that MediaPipe provided the best results. Hence MediaPipe is used for pose extraction to extract the person's pose in the image. When all the landmark points are connected, a human pose is a set of coordinates describing a person's pose. MediaPipe uses a high-fidelity pose-tracking model that extracts approximately 33 different landmarks. This model was used since it is a widely popular model for pose extraction. First, the MediaPipe model is declared with minimum detection confidence of 0.5 and minimum tracking confidence of 0.5. The model returns each key point as a list of

the x-axis, y-axis, z-axis, and visibility values. These values are flattened using the flatten() function and concatenated into a single array. The array contains 132 elements. MediaPipe also has functions that help draw the key points over the image. This model can also be used when more than one person is in the frame.

3.6 Model Training

The Images in the dataset are resized into common values. The images are then marked into five categories. If the person turns behind, then the image is marked as 0. If the person looks down to cheat, then the person is marked as 1. If the person looks into his hands, the image is marked as 2. If the person looks to the left, the image is marked as 3. Normal behavior is marked as 4. If the person looks to the right, the image is marked as 5. The pose arrays for all the images in the dataset are extracted and stored in a list. The dataset is divided into training and testing sets according to the 80:20 ratio. After exclusive testing, it was found that ExtraTrees Classifier and Linear Discriminant Analysis (LDA) provide the best result. Both models were evaluated, and the default parameters gave the best results. In the Results section, both the models were compared with the SVM model with Linear Kernel using classical parameters.

4 Result Analysis

The proposed model achieves modest results in all situations with varied lighting. The figures 4 - 7, below prove that the model can accurately detect head movements.

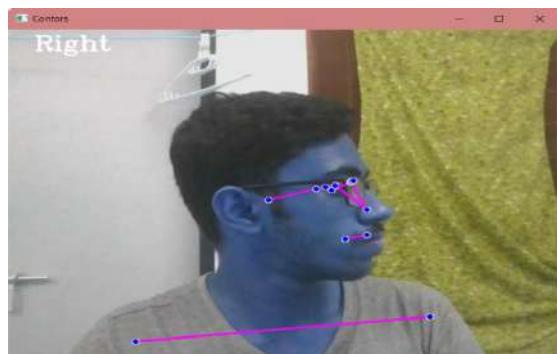


Fig. 4. Looking right



Fig. 5. Looking left



Fig. 6. Cheating through notes written on Hand



Fig. 7. Looking down

The model also exhibit exceptional results in the testing dataset with an accuracy of 95.9% using the ExtraTrees Classifier as shown in table 1 and LDA as shown in table 2. The classification reports are shown in figure 8 and 9 respectively.

Table 1. Evaluation Matrix for ExtraTree Classifier.

Parameter	Value (in %)
Precision score	95.9
Recall score	95.9
F1 Score	95.9

Table 2. Evaluation Matrix for LDA.

Parameter	Value (in %)
Precision score	95.1
Recall score	95.1
F1 Score	95.1

	precision	recall	f1-score	support
0	1.00	1.00	1.00	21
1	0.90	1.00	0.95	19
2	0.96	1.00	0.98	26
3	0.95	1.00	0.98	21
4	0.94	0.88	0.91	17
5	1.00	0.85	0.92	20
accuracy			0.96	124
macro avg	0.96	0.96	0.96	124
weighted avg	0.96	0.96	0.96	124

Fig. 8. Classification report for ExtraTree Classifier

	precision	recall	f1-score	support
0	1.00	0.95	0.98	21
1	0.95	1.00	0.97	19
2	0.96	1.00	0.98	26
3	0.91	1.00	0.95	21
4	1.00	0.82	0.90	17
5	0.90	0.90	0.90	20
accuracy			0.95	124
macro avg	0.95	0.95	0.95	124
weighted avg	0.95	0.95	0.95	124

Fig. 9. Classification report for LDA

If any person moves or changes their pose to cheat, the model will first alert the invigilator and simultaneously store the name and time at which the student attempted to cheat. Moreover, the model also stores all the frames for future reference units.

A common trend for this problem was that the models which could perform

feature selection extremely well yielded the best results. Also, models that could handle noisy data were more likely to generate good results. It is also known that the feature vectors have a certain degree of noise. This noise is why the ExtraTree classifier achieved the best results since it is better at dealing with noisy features. Also, it is a Tree-based method, which is more robust. ExtraTree uses randomization to decrease the variance compared to more classical models, such as random forest and decision tree models.

4 Conclusion

This paper proposes a model to help the proctors during the examination using computer vision. The model helps ensure examinations' fairness and solve traditional invigilators' problems. The proposed model warns the students and provides evidence if further investigations require. A robust pose detection system is proposed as the accuracy of the model is deeply affected by the accuracy of the pose detection system. The model can withstand adversarial attacks and make the model more resilient to cyber-attacks. The model was deployed in the cloud to ensure safety and for faster computations. More audio-based detection algorithms can also be implemented to catch students cheating through oral means.

References

1. S. L and C. S. Christopher: Video Surveillance using Deep Learning - A Review. In: International Conference on Recent Advances in Energy-efficient Computing and Communication, pp. 1-5. IEEE(2019)
2. MD Jibul Hoque, Md. Razu Ahmed, Md. Jashim Uddin and Muhammad Mostafa Amir Faisal: Automation of Traditional Exam Invigilation using CCTV and Bio-Metric. International Journal of Advanced Computer Science and Applications. 11(6), 392-399 (2020)
3. Y. Fang, J. Ye and H. Wang: Realization of Intelligent Invigilation System Based on Adaptive Threshold. In: 5th International Conference on Computer and Communication Systems, pp. 201-205, IEEE(2020)
4. S.-R. Ke, H. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi: A Review on Video-Based Human Activity Recognition. Computers, 2(2), 88–131 (2013)
5. Vishwakarma, S., Agrawal, A.: A survey on activity recognition and behavior understanding in video surveillance. Vis Comput, 29, 983–1009 (2013)
6. Douglas Attoh Odongo, Eric Agyemang, John Boulard Forkuor: Innovative Approaches to Cheating: An Exploration of Examination Cheating Techniques among Tertiary Students. Education Research International, 2021, 1-7 (2021)
7. Z. Cao, T. Simon, S. -E. Wei and Y. Sheikh.: Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1302-1310, (2017)
8. Suryadevara, Nagender.: Human Pose Estimation in the Browser. In: Beginning Machine Learning in the Browser. Apress, Berkeley, CA.(2021)

9. M. A. Samir, Y. Maged and A. Atia: Exam Cheating Detection System with Multiple-Human Pose Estimation. In: IEEE International Conference on Computing, pp. 236-240, (2021)
10. S. Maniar, K. Sukhani, K. Shah and S. Dhage: Automated Proctoring System using Computer Vision Techniques. In: International Conference on System, Computation, Automation and Networking, pp. 1-6, (2021)
11. J. Nishchal, S. Reddy and P. N. Navya: Automated Cheating Detection in Exams using Posture and Emotion Analysis. In: IEEE International Conference on Electronics, Computing and Communication Technologies, pp. 1-6, IEEE (2020)
12. Kohli, S. & Jannaj, Y. & Maanan, Mohamed & Rhinane, Hassan: Deep Learning: New Approach For Detecting Scholar Exams Fraud. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. XLVI-4/W3-2021, 103-107 (2022)
13. Riaz, Hamza, Uzair, Muhammad, Ullah, Habib & Ullah, Mohib: Anomalous human action detection using a cascade of deep learning models. In: Proceedings of European Workshop on Visual Information Processing, EUVIP, IEEE, US (2021)
14. Mahmood, Fatima & Arshad, Jehangir & Othman, Mohamed & Hayat, Muhammad & Bhatti, Naeem & Jaffery, Mujtaba & Rehman, Ateeq & Hamam, Habib: Implementation of an Intelligent Exam Supervision System Using Deep Learning Algorithms. Sensors. 22, 6389 (2022)
15. Razan Bawarith, Dr. Abdullah Basuhail, Dr. Anas Fattouh and Prof. Dr. Shehab Gamalel-Din: E-exam Cheating Detection System. International Journal of Advanced Computer Science and Applications, 8(4), 2017.
16. T. Nakashima and Y. Yabuta: Object Detection by using Interframe Difference Algorithm. In 12th France-Japan and 10th Europe-Asia Congress on Mechatronics, pp. 98-102 (2018)

Dynamic Wireless Charging System for Electric Vehicles based on Segmented Zinc Alloy Plates

Anurag A Gadgil¹, Arya Bairoliya¹, *Febin Daya J.L², and Balamurugan P²

¹VIT Bhopal University

²Vellore Institute of Technology – Chennai Campus

²*febindowayajl@vit.ac.in

* Corresponding Author

Abstract. This paper proposes a dynamic wireless charging system based on segmented alloy plates for charging electric vehicles on the go. With the rise and development of electric vehicles (EV), there is an increase in demand for charging infrastructure for batteries. Wireless charging on EV eliminates the need for wires and connectors to charge the vehicle, thereby making the charging easy and hassle-free. Dynamic charging is an incremental addition over static wireless charging where the battery in EV is charged when the vehicle is in motion. This also reduces the battery capacity required, thereby making EV much cheaper. Considering the EV sector, a dynamic charger that will charge the vehicle on the go will not only eliminate the old conventional need for stopping for long hours but also increase travel efficiency to a whole new level. In this proposed work, the dynamic charging system is designed and simulated using matlab. The thermal modelling and analysis of the charger are done using solid works and the results are presented.

Keywords: Dynamic charging, Resonance frequency, Charge plates, Thermal analysis.

1 Introduction

Electric vehicle adoption is gaining momentum in recent days. The sale of 2 wheelers and 4 wheelers EV are on the rise as per the parivahan seva data by Ministry of road transport and highway, India. However, deployment of public charging infrastructure is not at the same pace when compared to EV sales. The rising electric car market fuels the demand for more convenient and dependable methods of charging electric vehicle[1]. The conventional charging method uses On board chargers or the high power chargers available in charging stations to recharge the battery in EV. In addition, incompatible plug receptacles generate complications among EV models, limiting the rapid adoption of EVs. Wireless charging is an alternative option to build this gap and fuel the rapid adoption of EV. Dynamic wireless charging allows the vehicle to be charged on the go [2-4]. This can reduce the battery capacity by 1/5th time which in turn contributes to vehicle efficiency (wh/km) and the overall cost of the vehicle. Dynamic wireless charging of electric vehicles could become a preferred method since it would enable power

exchange between the vehicle and the grid while the vehicle is moving ubiquitously. Hence there was a growing interest in charging of vehicles on the move. In dynamic charging, onboard inductive units are used to pick up charge from power sources buried in the road or located above the surface. Dynamic charging has the inherent advantage of addressing some of the major challenges of EV battery charging such as charging time, battery capacity, battery size and travel range. The schematic of dynamic power transfer system for EV is shown in Fig.1.

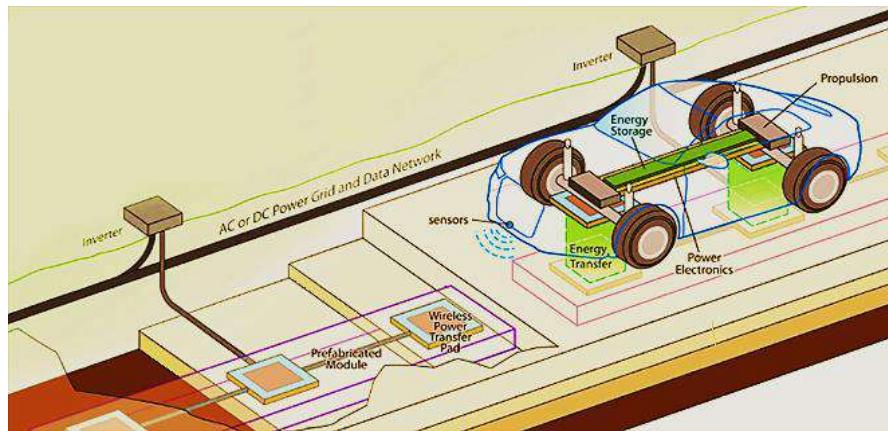


Fig. 1. Schematic of dynamic wireless power transfer system for EV

2 Literature Review

Research articles in the area of dynamic wireless charging of EV are also reported in recent literature. Omar et.al proposed a technique for transferring power through a wireless medium for a considerably larger distance. The proposed method also included a dynamic positioning system which enhanced the efficiency of power transfer between the coils. [5]. The vehicle position tracking and self-alignment feature further enhanced the overall performance of the wireless charging system. Hutchinson et. al studied the impact of dynamic wireless charging systems in terms of vehicle cost, range, battery life, user convenience and charging infrastructure for EV market penetration[6-8]. This study also gave insight into the influential parameters of dynamic wireless charging. Some of the recent notable improvements in dynamic charging were presented. Dai et.al, proposed a multi-excitation unit configuration for EV dynamical charging [9]. This method will be useful for improving the power capacity of the charger coils. The high power capability of the primary and secondary coil is achieved by using more excitation units in the area where coils are embedded, thereby reducing the voltage stress on

the converters. Azad et. al, presented a detailed study on the different types of coil structures used for dynamic power transfer. The coil size, efficiency and power transfer capability are compared for various cases. Dynamic charging needs separate road / pathway wherein there are provisions for embedding the coils, coil connections with supply system, mechanism for avoiding foreign object interventions [10-11]. Sooraj et.al has proposed a dual side control mechanism for wireless charging of electric vehicle. High power transfer efficiency was achieved using this technique [12]. They have also proposed an algorithm for estimating the optimal frequency for maximum power transfer under misalignment condition. The optimal frequency maintains resonance between the coils, thereby ensuring maximum power transfer [13]. Sooraj et. al have proposed a dynamic model for wireless power transfer system when the coils are loosely coupled [14]. They have also proposed a high gain LCL topology for efficient wireless transfer. The high gain topology eliminated the need of intermittent DC-DC converter in the wireless power transfer system [15]. A detailed review on power converter topologies is presented by Sooraj et.al. The advantages of different converter topologies, power level, operating frequency, power transfer efficiency and misalignment analysis are studied in detail [16]. The paper then facilitates a comparison of the various inductive pads, rails, and compensation systems developed to date. Additionally highlighted are the static and dynamic charging methods and their traits. The function and significance of different types of power electronics and converters utilized in applications are explained. Additionally discussed are the batteries and their management systems as well as several WPT-related issues. A variety of trades are investigated, including cyber security, economic consequences, health and safety, identification of foreign objects, and effects on the distribution grid.

In the proposed work, a wireless power transfer system for EV is presented. It offers a significant advantage of power transfer, to prevent physical touch and being plugged in regularly. The coil shape, dimensions, gap between coil pads and magnetic core material used plays a critical role in deciding the performance of the wireless power transfer system. In the case of dynamic power transfer systems, a large number of coil pads are embedded on the road surface which acts as the primary coil for power transmission. A stepped coil pad arrangement is proposed in this work for dynamic power transfer. The coils are arranged using segmented alloy plates. The proposed design is simulated and the thermal aspects are studied and presented.

2 System Modeling and Design

The schematic diagram of the wireless charging system is shown in Fig. 2. The single-phase AC input is given as input to the rectifier. After rectification, the inverter switches at a high frequency and generates a high-frequency signal, which is given as input to the primary coil pads. In the primary coil, the compensating networks are present, to make the circuit operate at the resonant frequency. The

secondary coil pads are connected through a high-frequency rectifier and filter to the load. The Li-ion battery is considered as load. The circuit parameters design using Suitable equations [4] are given in Table1.

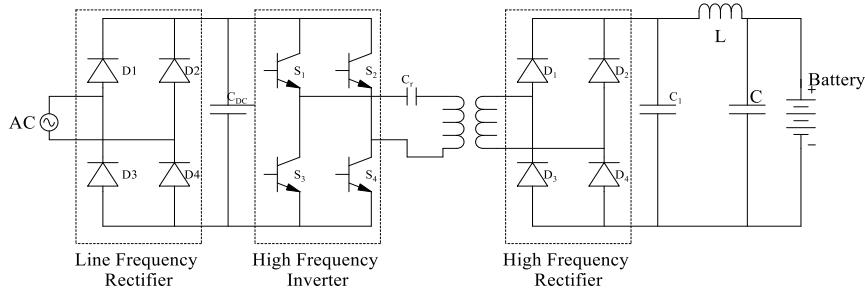


Fig.2. Circuit diagram of wireless battery charger

Table 1. Coil parameters.

Component	Quantity
Number of Turns	25
Coil diameter	760mm
Primary coil resistance	0.0714Ω
Secondary coil resistance	0.1428Ω
Mutual Coupling (k)	0.1 – 0.5
Primary coil inductance	0.07343 mH
Secondary coil inductance	0.07343 mH
Resonance Frequency	41.53 kHz

3 Thermal Modeling and Design

The CAD Model of the charger pads for dynamic charging is prepared using Solidworks. The idea behind this wireless charger is that it be installed 2 inches below the road and charge the vehicles dynamically. The coils are arranged in such a way that the magnetic leakage in minimum so that maximum flux linkage yields maximum efficiency [17]. The power transfer efficiency and packaging efficiency are taken into considerations while arranging the coils. The plates are segmented with each segment having a distance of 1 m between them. The width of the plate as whole is 3.75m which matches the standard size of road lane. The length of the plate can be varied as per requirement. The advised length of the single plate should not exceed 15 m as this might cause an issue for current dissipation in whole. Given the charging plate will be installed under the road which generally is made up of tar or concrete it must endure higher temperatures and cyclic stress. The constant movement of vehicles over it will induce several tons of force every second. On top

of that it must endure blistering temperatures. For the same reason, the segmented charging plate was tested in Solidworks at a maximum environmental temperature of 100 Degree Celsius [18]. The internal temperature of the plate was not taken into consideration since zinc is a moderately good conductor of heat and the movement of electrons will not cause it to rise to temperatures above 30-40 Degrees. The 3D model of the charger installed under the road is shown in Fig. 3.

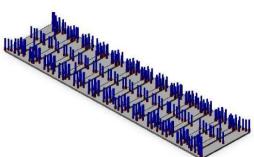
Solid Bodies		
Document Name and Reference	Treated As	Volumetric Properties
Fillet1 	Solid Body	Mass:80,103.8 kg Volume:12.1369 m^3 Density:6,600 kg/m^3 Weight:785,017 N

Fig. 3. The 3D Model of the charger that will be installed under the road.

3.1 Material properties and Thermal load

The charger material properties simulationis shown in Fig. 4. Zinc was the primary candidate since it is a good conductor of electricity and a moderate conductor of heat. Being highly anisotropic in nature it had to be alloyed with other materials like aluminum which increase durability, rigidity and performance [19]. It s a sought after alloy above aluminum, magnesium and bronze. Alloyed with aluminum it tends to follow the Linear Elastic Isotropic model. This models assumes a linear relation between between the stress ans stratin there by making the model simple

and easy to analyse. This model shows that the properties of the alloy do not vary with direction and the properties are generally predictable at different stresses at different load axis like X,Y and Z-Axis. The ultimate tensile strength of the zinc alloy is 352 MPa and the electrical conductivity of the same is 25 Siemens per meter [11]. Fig. 5 shows the simulation shows the heat is applied on the faces of the segmented plates

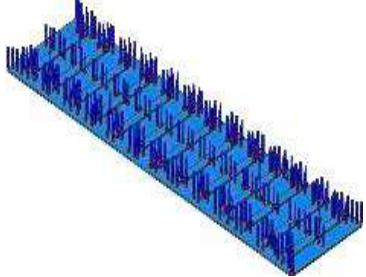
Model Reference	Properties
	Name: Zinc Alloy 7; AG40B; Zn-4Al-0.015Mg Model type: Linear Elastic Isotropic Default failure criterion: Unknown Thermal conductivity: 113 W/(m.K) Specific heat: 418.7 J/(kg.K) Mass density: 6,600 kg/m^3

Fig. 4. Material Properties of the Charger.

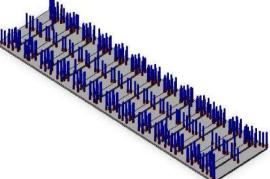
Load name	Load Image	Load Details
Temperature		Entities: 14 face(s) Temperature: 373 Kelvin

Fig. 5. Application of heat on the faces of the charger.

4 Results and Discussion

The wireless charging system is simulated using Matlab /Simulink. The simulation results are shown in Fig. 6 . The wireless charger was able to charge the Li-ion battery and the SoC of the Battery started increasing from 80%. The simulation results of the charger for different temperature ranges are shown in Fig. 6. In the simulation model a single-phase AC input of 230V, 50Hz is given as input to the rectifier. After rectification, the inverter switches at a high frequency and generated a high-frequency signal, which is given as input to the primary coil pads which are having self-impedance parameters as 0.0714ohms and 73.43uH.

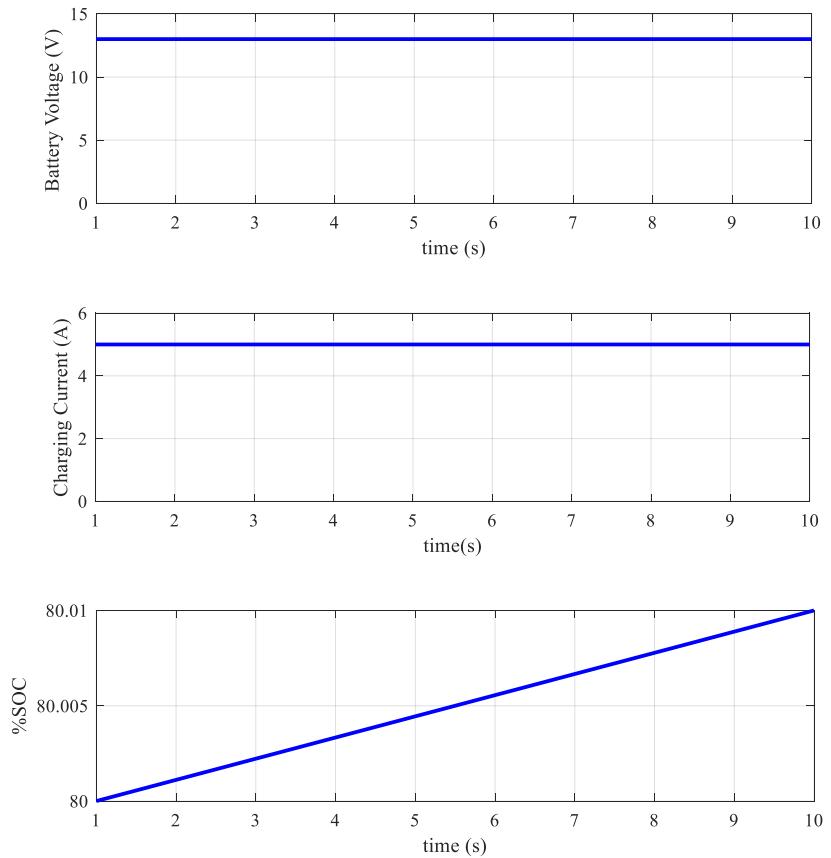


Fig. 6. Output Graph - This graph on MATLAB shows the ripples generated by the charger while charging resulting in lower frequency and accurate charging.

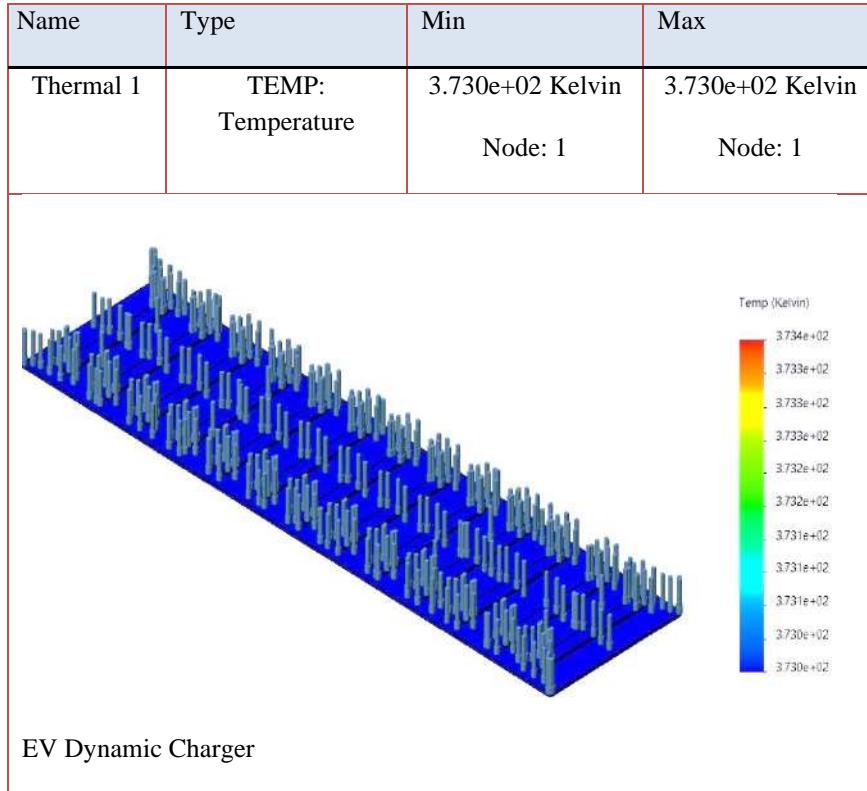


Fig. 7. Charge distribution across the charger plates.

In the primary coil, the compensating networks are present, to make the circuit operate at the resonant frequency. The secondary coil pads are having self-impedance parameters as 0.0714Ω and $73.43\mu\text{H}$ which are connected through a high-frequency rectifier and filter to the load. The mutual impedance parameters of the primary and secondary coils are 0.142Ω and $14.6\mu\text{H}$. The Li-ion battery is considered as load, which is having the specifications as 12V, 20Ah. The voltage, current and SOC is measured through scope in MATLAB. The output shows that the battery is charging to a maximum voltage of 13 V, the corresponding flow in current thereby the SOC is increasing that depicts that the battery pack is charging. The plate temperature is tested at various instances and the results are consolidated. The simulation results are shown in Fig. 7. The analysis output was satisfactory and as per requirement. The plate maintains its original temperature and does not result in overheating which could have caused a problem.

5 Conclusion

This paper proposed the dynamic charger with segmented alloy plates for Electric vehicle. The charger was designed and simulated. The SoC of the li-ion battery started increasing, when the secondary pad was in close proximity with the primary. The SoC of the battery raised at a faster rate when the gap between the coil pads are lesser. The vehicle range and overall performance is improved to a greater extent. A suitable control algorithm is to be deployed to perform dual side control of the overall system. Preliminary simulations show that applying some form of intelligent control technique can further improve the performance.

Future research will focus on the electromagnetic interference (EMI) that the transmitter coils emit. The transferrable power in real use must be increased with further EMI reduction. The efficiency of the system can be increased to ensure more amount of power transfer. Future work also includes a study on the bidirectional power flow between the infrastructure and the on-board battery, which is necessary to support a vehicle-to-grid (V2G) concept.

References

1. Song, S., Dong, S., Zhang, Q.: Receiver Current-Stress Mitigation for a Dynamic Wireless Charging System Employing Constant Resistance Control. *IEEE Trans. Power Electron.* 36, 3883–3893 (2021)
2. Omar, N., Santos, E.D., Yago, A.: Wide Range Highly Efficient Dynamic Wireless Power Transfer System. In: *IEEE Transportation Electrification Conference & Expo (ITEC)*, IEEE Press, New York (2021)
3. Hutchinson, L., Waterson, B., Anvari, B., Naberezhnykh, D.: Potential of wireless power transfer for dynamic charging of electric vehicles. *IET Intelligent Transport Systems*. 13, 3 – 12 (2019)
4. Qui, C., Chau, K.T., Liu, C.: Overview of wireless charging technologies for electric vehicles. *J. Asian Electric Vehic.* 12, 1679–1685 (2014)
5. Liu, C., Chau K.T., Qui, C., Lin, F.: Investigation of energy harvesting for magnetic sensor arrays on Mars by wireless power transmission. *J. Applied Phy.* 115, 1 – 3 (2014)
6. Zhang, Z., Chau K.T., Qui, C., Lin, C.: Energy encryption for wireless power transfer. *IEEE Trans. Power Elct.* 30, 5237–5246 (2015)
7. Liu, C., Chau K.T., Zhang, Z., Qui, C., Lin, F., Ching, T.W.: Multipleresistor wireless power transfer for magnetic sensors charging on Mars via magnetic resonant coupling. *J. Applied Phy.* 117, 17A743:1 – 3 (2015)
8. Chen, W., Liu, C., Lee, C.H.T., Shan, Z.: Cost-effectiveness comparison of coupler designs of wireless power transfer for electric vehicle dynamic charging. *Energies*. 11, 906 – 918 (2016)
9. Dai, X., Jiang, J.C., Wu, J.Q.: Charging Area Determining and Power Enhancement Method for Multi excitation Unit Configuration of Wirelessly Dynamic Charging EV System. *IEEE Tran. Ind. Electron.* 66, 4086 – 4096 (2019)
10. Liu, C., Chau K.T., Zhang, Z., Qui, C., Li, W., Ching, T.W.: Wireless power transfer and fault diagnosis of high-voltage power line via robotic bird. *J. Applied Phy.* 117, 17D521: 1 –

4 (2015)

11. Jiang, C.Q., Chang, K.T., Liu, C., Han, W.: Wireless dc motor drives with selectability and controllability. *Energies.* 10, 1–15 (2017)
12. Sooraj, V., Febin Daya, J.L.: Compact Pulse Position-Based High Efficient IPT System for Wireless Charging of EV. *IET Power Electronics.* 13, 85–96 (2020)
13. Sooraj, V., Febin Daya, J.L.: Estimation of Optimal Operating Frequency for Wireless EV Charging System under Misalignment. *Electronics.* 8, (2019)
14. Sooraj, V., Febin Daya, J.L.: A Large Signal Dynamic Model of Loosely Coupled IPT System for EV Based on Laplace Phasor Transform. In: *IEEE Power Electronics, Drives and Energy systems Conference*, pp. 1-6. IEEE Press, New York (2018)
15. Sooraj, V., Febin Daya, J.L.: A High Gain LCL Architecture Based IPT System for Wireless Charging of EV. *IET Power Electronics.* 12, 195–203 (2019)
16. Sooraj, V., Febin Daya, J.L., Mohan Krishna, S., Williamson, S., Ramani, K., Lila., I.I.: Role of Power Converters in Inductive Power Transfer System for Public Transport—A Comprehensive Review. *Symmetry.* 14, (2022)
17. Bosshard, R., Kolar, J.W.: Inductive Power Transfer for Electric Vehicle Charging: Technical Challenges and Tradeoffs. *IEEE Power Electron. Mag.* 3, 22–30 (2016)
18. Knecht, O., Kolar, J.W.: Performance Evaluation of Series-Compensated IPT Systems for Transcutaneous Energy Transfer. *IEEE Trans. Power Electron.* 34, 438 – 451 (2019)
19. Mohamed, A.A.S., Meintz, A., Schrafel, P., Calabro, A.: Testing and Assessment of EMFs and Touch Currents From 25-KW IPT System for Medium-Duty EVs. *IEEE Trans. Veh. Technol.* 68, 7477–7487 (2019)

Iterative Refinement vs Generative Adversarial Networks for Super-Resolution toward Licence Plate Detection.

Alden Boby^{1[0000-0002-6483-4253]}, Dane Brown^{1[0000-0001-7395-7370]}, and James Connan¹

Rhodes University, Department of Computer Science, Grahamstown, South Africa
`boby.alden128@gmail.com d.brown@ru.ac.za j.connan@ru.ac.za`

Abstract. Licence plate detection in unconstrained scenarios can be difficult because of the medium used to capture the data. Such data is not captured at very high resolution for practical reasons. Super-resolution can be used to improve the resolution of an image with fidelity beyond that of non-machine learning-based image upscaling algorithms such as bilinear or bicubic upscaling. Technological advances have introduced more than one way to perform super-resolution, with the best results coming from generative adversarial networks and iterative refinement with diffusion-based models. This paper puts the two best-performing super-resolution models against each other to see which is best for licence plate super-resolution. Quantitative results favour the generative adversarial network, while qualitative results lean toward the iterative refinement model.

Keywords: Character Recognition · Diffusion Probabilistic Models · Licence Plate Recognition · Object Detection · Super-Resolution

1 Introduction

The number of vehicles on the road is rapidly increasing, and the demand for vehicles creates a need for more robust road management and traffic regulation. There is a need to monitor vehicles and introduce regulations which combat congestion on the roads and promote safety for pedestrians and drivers alike. Advances in technology can supplement these rules and regulations already in place. These are referred to as intelligent transport systems (ITS) [21].

Vehicles are required by law to have a licence plate. Although it maintains a uniform shape and size from vehicle to vehicle, these licence plates have a set of characters that are uniquely arranged and can be used to identify: the owner, the registration, the class of the vehicle and other related information. This information is specifically important to government agencies such as traffic police and law enforcement as it can aid their daily jobs.

The need to track vehicles extends past law enforcement, although them being the primary users, insurance companies, and customers could benefit from

these advancements when investigating a claim. Provided that there is footage of an accident, the company could identify the vehicle in the wrong through the licence plate and pass this information to the traffic enforcers, who can identify the vehicle's owner.

A major obstacle with licence plate recognition is the medium on which the data has been captured. Images or videos could have a low resolution, making it hard to identify objects of interest. This is especially true with licence plates, as they typically occupy a tiny portion of an image. This results in a loss of information or distortion of the image, making characters in the image or frame unidentifiable. Figure. 1 demonstrates this problem.



Fig. 1: The highlighted licence plates only occupies a small portion of the entire image.

A myriad of problems makes licence plate detection difficult besides a low resolution. Once a licence plate has been extracted, it can only be given context once the characters on the licence plate are correctly identified. One incorrect character can render the whole licence plate invalid. A solution to this problem is super-resolution (SR) to improve the quality of an image before further processing it for character recognition.

2 Super-Resolution

Super-resolution is applicable across many domains [1]. In the licence plate detection field, it would be beneficial to use this upscaling method to improve accuracy when detecting characters on a licence plate. The advantage of using a fully computational model is that expensive hardware costs are mitigated. For example, police vehicles are fitted with expensive camera systems consisting of

telephoto and infrared cameras specifically configured to capture licence plates. Although that is a working solution, it would be beneficial to create a software-based solution with comparable performance that can be deployed as a more affordable solution, whether for a government organisation or a private entity.

This is where SR can be applied. Unlike standard upscaling methods like bi-linear or bi-cubic upscaling, the use of SR improves the quality of an image without prior knowledge of the ground truth image. SR produces superior-quality images that have been upscaled with standard sharpening and deblurring filters. Figure 2 demonstrates the superiority of SR as an upscaling method.



Fig. 2: Deep-learning methods produce much sharper upscaled images [18].

This is perfect for what is trying to be achieved as it eliminates specialised hardware. Super-resolution models work through partially supervised machine learning; they are fed images mapped to each other to inform the system. The models then extrapolate the information from the training samples to upscale unseen images; this is referred to as blind SR. Super-resolution can be achieved through deep learning models, specifically generative adversarial networks (GANs) and diffusion models, which will be the focus of this paper. GANs do not perform exceptionally well on finite details like text. The use of an alternative, such as iterative refinement, may prove to be better.

2.1 Diffusion Probabilistic Model

A diffusion probabilistic model, or diffusion model for short, works with an iterative process that performs particularly well for image generation. For machine learning models such as diffusion probabilistic models and GANs alike, a new high-resolution image is essentially a newly generated image. Hence, the models can be further tweaked, allowing them to be exploited for SR.

The diffusion model is a Markov chain that slowly adds noise to an image [7]. An input image is taken, and Gaussian noise is applied to it, then, it is denoised in an iterative process creating a high-resolution image as the resulting output. Figure 3 shows an example of this process and how it takes place, each time-step, t , relies on the previous step creating a less noisy image at each time step (x_{t-1}, x_{t-2}, \dots). Generally, the more iterations the model performs, the better the quality of the final output image.



Fig. 3: Each image in the progression is a less noisy version of x_t , x_{t-1} .

A current limitation of these diffusion models is their ability to run in real time. They have a linear time complexity, and the more iterations the model has to go through, the longer the image will take to produce.

2.2 Generative Adversarial Networks

Generative adversarial networks have been the most popular model for image generation prior to advancements in the diffusion model space. A GAN is comprised of two networks, which are a discriminator and a generator. To train a GAN, these networks work together in an opposing manner to improve model performance. The generator generates some image, and the discriminator is supposed to discern whether the output from the generator is a fake or an actual image [3]. Depending on the outcome, the weights of the model will be updated according to the loss function. Eventually, the generator becomes good enough to fool the discriminator each time, allowing the generator to be used on its own to produce its own images or upscale images in the context of this research. GANs are unsupervised machine learning models that can be supplied with unlabelled training data. A GAN needs to be able to produce a diverse set of images. GANs can suffer from mode collapse, which causes them to produce the same or similar image each time which is undesirable if one wants to upscale or create more than one image [17]. Mode collapse can occur during the training process, causing the generator to link several input points to one output, resulting in similar output images for a range of varied data – which is undesirable.

3 Object Detection

Character recognition is an important stage for licence plate recognition as this is the stage where data turns into information that can be used to look up vehicles that have been recognised. The use of an object detector can be used to detect these characters with relatively high accuracy. Supplemented by the use of SR, the detection rate of the models should have a noticeable increase, especially on images where the quality of the image affects the look of certain characters. The You Only Look Once (YOLO) object detector can be used for this.

4 Related Studies

The first step in licence plate recognition requires extracting a licence plate. YOLO has been successful in this regard, being used as the go-to deep-learning

method, surpassing other object detectors such as Region-based convolutional neural networks (R-CNN), detecting an object and its class in one pass [5]. Many authors such as Lee *et al.* [9], Silva et al. [16], and Boby *et al.* [2] all achieved success through the use of YOLO for licence plate extraction. Silvia and Jung further improved the performance of YOLO by creating a specialised architecture heavily based on YOLO called the warped planar object detection network (WPOD-net) [15]. This model introduced bounding parallelograms, which are more suited to the varied angles present in real-world data.

Convolutional neural networks (CNNs) are at the base of most computer vision tasks, originally creating SR images through end-to-end mapped training data. The Super Resolution Convolutional Neural Network (SRCNN) was one of the first successful deep learning models for super-resolution [1]. The model surpassed the quality of images upsized with bicubic upscaling. [6].

GANs were quickly adopted and used in place of CNNs due to their superior image quality. In the current literature GANs have been the more popular choice for SR with licence plates. Lee et al. [9] used GANs to upscale low-resolution licence plates and successfully reduced false positives at the character recognition stage. Similarly, Boby et al. [2] succeeded by using an ESRGAN to enhance low-resolution licence plate images resulting in increased character recognition accuracy. These positive results from using SR mean that trying other SR methods may further accelerate progress in this field.

Diffusion models have recently started gaining traction for a number of applications such as image synthesis [12,10,11] as well as SR [13]. The new advancements in diffusion models mean there needs to be more application in the licence plate detection field. This may be due to real-time requirements of licence plate detection, and these diffusion models can not upscale or generate an image in real time. Regardless of this, diffusion-based SR models have very impressive results. Figure 4 shows an image generated by SR3, a diffusion model developed by a team at Google. This is still advancing as most diffusion models have only been released in 2022; this includes DALLE-2 [10], Imagen [12], and Stable Diffusion [11]. Although Imagen and DALLE-2 are explicitly built for image synthesis, they have a SR component within them, demonstrating that diffusion models are an alternative to GANs for SR. Figure 4 shows the potential of SR with diffusion-based models.

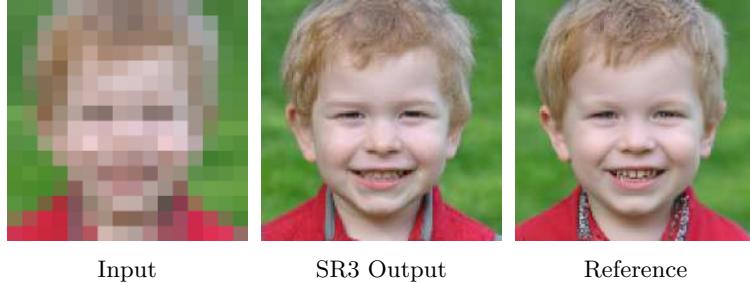


Fig. 4: The close similarity between the synthetic image and the ground truth image present a case for diffusion models as a substitute for SR. [13].

Dhariwal et al. [4] present a paper on how diffusion models outperform GANs for image synthesis. Describing GANs as the current state-of-the-art regarding image generation and how diffusion models outperform them in the same task. In the case of SR, the output from the model is still synthesised and should carry over to that space. Other authors share the same sentiment and compare the performance of their diffusion models to GANs reporting more satisfactory results from the diffusion models. [11,13].

It is difficult to quantify image quality, and finding a suitable measure to evaluate the output from a SR model is necessary. This inspection can be done visually as well as mathematically. A mathematical representation makes assessing image quality difficult, as qualitative results allow for a way to measure a model's performance [20]. Mean squared error (MSE) can be used to measure the quality of an image, but as described by Wang et al. [19]. MSE for an image does sufficiently represent the human perception of an image. For example, Figure . 5 shows the same image with an MSE of 309; however the images vary drastically in quality.



(a) Noticeable degradation and colour banding in the image. (b) No easily noticeable flaws in the image.

Fig. 5: The images are perceivably different quality, however, have the same MSE [19].

This demonstrates that MSE is not the best measure to evaluate images. To date, literature is still using the peak-signal-to-noise ratio (PSNR) and structural similarity index (SSIM) to measure the quality of an image. Looking back at Figure 5, although they have the same MSE, their SSIM is different, showing that it is a much better metric to use. However, visual inspection is still helpful as no metric can fully quantify human perception when viewing images [14]. This informs the methodology that a human should qualitatively analyse some samples to compare the outputs from a diffusion model and a GAN.

Specifically for SR, the SR model tries to re-imagine an image based on the training data supplied. Metrics like SSIM and PSNR measure the similarity between images with image synthesis the SR output will always differ from the original. In this case, the goal of the SR is to make characters appear clearer so that they can be detected clearly [2]. Regardless, SSIM is currently the best way to evaluate image quality considering human perception [14].

With recent literature exploring its use, YOLO is a strong candidate for character recognition. Kim *et al.* [8] used YOLOv2 as a substitute for OCR and found that YOLO performed much better for character recognition. Silvia used CR-NET and observed a similar outcome, with a 1.71% increase in accuracy from using the deep-learning model compared to a commercially available system [16]. Insight from these papers informs the experiments conducted in this research, making YOLO a suitable candidate for character recognition.

5 Proposed Work

This paper uses a diffusion model implementation based on the super-resolution via iterative refinement model created by Google [13]. The implementation in use does not match the performance levels of those shown in the research paper as it is a reconstruction of the model based on information from the publication, which omitted some details to create the exact system. The model ¹, however, is sufficient enough to use. The second model is the Real-ESRGAN [18], a state-of-the-art GAN-based SR model.

The SR model is trained with relevant LP data to perform better in the licence plate domain. The majority of SR models are tuned to work on faces; supplementing it with specific training data should boost performance for licence plates as certain features on licence plates are not found in natural scenes.

The models will take a low-resolution 64×64 image and then upscale it by a factor of 8 to 512×512 . Both models will be supplied with the same dataset so that the output from the models is directly comparable. Diffusion models require the input image to be the same size as the output image. So the diffusion model is supplied with a 64×64 image upscaled to 512×512 with bicubic interpolation as an intermediate prepossessing step before being upscaled via super-resolution. Figure 6 shows a high-level overview of the methodology.

¹ <https://github.com/Janspiry/Image-Super-Resolution-via-Iterative-Refinement>

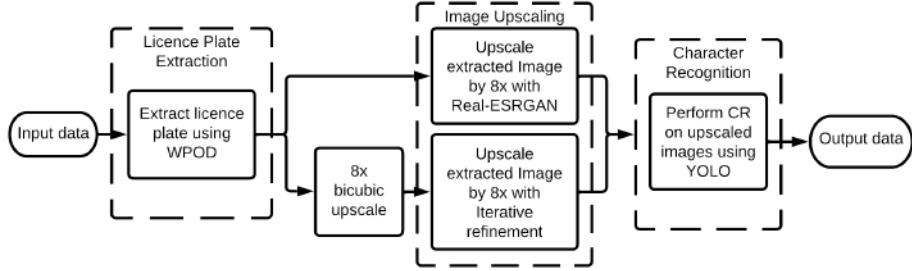


Fig. 6: High level overview of the proposed LPR system.

The high-resolution output from both models is compared both quantitatively using performance metrics specific to image quality and qualitatively through visual inspection.

5.1 Measuring Image quality

The paper will use both qualitative and quantitative methods to inspect SR outputs from two SR models: the Real-ESRGAN and a diffusion model. The quantitative metrics discussed in the related studies section are described in the following sections.

SSIM

Structural Similarity is a metric used to measure the similarity between two images. By measuring the SSIM of a ground truth high-resolution image compared against a SR image, it is possible to estimate how well the upscaling model performed. SSIM is the metric closest to describing human perception.

PSNR

PSNR is an image quality assessment metric that uses signals within an image to assess to what extent noise distorts an image. The ratio is used to compare the similarity between two images. PSNR is measured in decibels, where a higher value is preferable. Equation 1 shows how to calculate PSNR.

$$\text{PSNR} = 10 \log_{10} (\text{peakval}^2) / \text{MSE} \quad (1)$$

Peakval represents the highest value in the input image (peak value).

5.2 Model Training

The super-resolution models were trained on a device with the following specs:

- GPU: 2080TI (11GB RAM)

- CPU: AMD Ryzen 9 3950X 16-Core (4.7 GHz)
- RAM: 128GB

Training the diffusion model takes a considerable amount of VRAM, so the batch size was set to one to allow the training to occur with the available amount of VRAM.

The model was trained on a set of one hundred images; diffusion models have the added benefit of being easier to train than GANs. The images contained licence plates with a variety of different text colours and background colours. Ensuring the training data is diverse directly affects the quality of the end results from the model. The model was trained for 60,000 epochs, and 2000 iterations were used for each epoch.

5.3 Test Models

A series of experiments will be conducted to compare the performance of the two super-resolution models and, ultimately, which model helps to improve character recognition the most.

Experiment 1

A quantitative evaluation of the SR images produced by both models using SSIM and PSNR, which are the current standard to measure an images quality.

Experiment 2

A qualitative analysis of SR images produced by both models as the PSNR and SSIM metrics cannot fully quantify human perception.

Experiment 3

Performing character recognition on images upscaled by the Real-ESRGAN and the diffusion model to analyse how well each model reconstructed characters.

6 Result Analysis

6.1 Experiment 1

Table 1: Image quality assessment scores for the SR models.

Metric	Real-ESRGAN	Diffusion Model
PSNR	22.086	16.876
SSIM	0.67072	0.34417

The PSNR and SSIM for both models are shown in Table 1. The Real-ESRGAN had better results for both SSIM and PSNR, with the difference between SSIM being almost double. Quantitatively the Real-ESRGAN performed better, meaning the results from this model should have a more significant effect on character recognition than the diffusion model. The following experiment will examine the extent to which the qualitative and quantitative results correlate.

6.2 Experiment 2

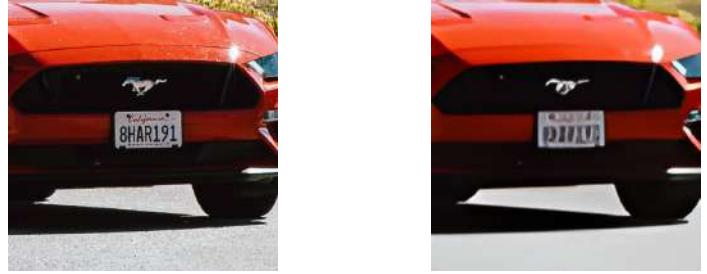
Figure 7 shows the SR image from the diffusion model with the lowest PSNR and SSIM. Upon visual inspection, the image is indeed the worst quality out of all the images output from the diffusion model. Conversely, the same image from the Real-ESRGAN has a much higher PSNR and SSIM values, but the licence plate in the image is still illegible. It appears colour takes high precedence with these evaluation metrics, as between the diffusion model output and the Real-ESRGAN output, the GAN has a more accurate colour when compared to the original image. In contrast, the diffusion model output lacks vibrance.



Fig. 7: The worst quality image from the diffusion model compared against the original HR image and the Real-ESRGAN output.

A limitation is that certain patterns become affected when an image’s resolution is too low. Patterns become interpreted differently when upscaled, producing an erroneous SR image, for example in Figure 7, the rear vents of the Porsche appear to collapse on each other, and that is then extrapolated to the SR and mismatching the ground truth. This often occurs with sets of parallel lines.

The Real-ESRGAN smoothes out fine details in an image giving the output a water or oil-painting look, more-so than resembling a natural image. This appearance makes it easily distinguishable to the human eye. Figure 8 demonstrates how the output from the Real-ESRGAN is missing some finer details that have been washed out.



(a) Original HR image. (b) Real-ESRGAN output.

Fig. 8: Image 8b has reduced fine details and is very smooth in appearance. Textures like grains in the road have been lost in the SR process.

The diffusion-based model produced some promising results but was very inconsistent. In some areas clearing up distortions and reforming unclear characters. Some instances produced images that have some Gaussian noise. The zoomed image in Figure 9 shows that the unclear characters have been cleared up after the diffusion process.

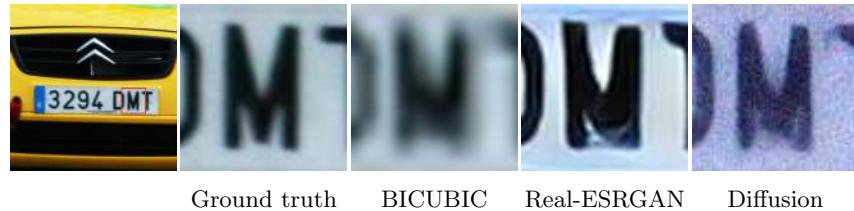


Fig. 9: A close-up at the letter ‘M’ to see how each model reconstructed characters on a licence plate.

Figure 9, demonstrates both the weak and strong points of the diffusion model. The Real-ESRGAN has made the character ‘M’ ambiguous and could be misinterpreted as an ‘N’, while the diffusion model has recovered some detail making the blurry character in the bicubic image look more like an ‘M’. However, the SR image from the diffusion model has visible noise, an issue that can be solved by increasing the number of iterations. However, the increase in quality from double the amount of iterations is negligible when the increase in time is considered. Figure 10 shows another example of how both models reconstructed characters.

Images with licence plates occupying a small portion of the image could not be restored by any model, the amount of information in such images was too little to extrapolate.



Fig. 10: The ‘S’ can be seen to have exaggerated features when upscaled with the Real-ESRGAN when compared to the more readable Diffusion model upscale.

6.3 Experiment 3

The upscaled images were used to perform character recognition with a trained YOLO model. The results are shown in Table 2. The best performance came from using the Real-ESRGAN as the upscaler, with precision and recall going up. A correlation between the quantitative results can be found here, as the Real-ESRGAN also had the highest scores when the SSIM and PSNR were measured. Although performing worse in all quantitative tests, the diffusion model still had some promising upscaled results.

Table 2: Character recognition results on upscaled images.

Upscaling Method	Precision	Recall
Bicubic	0.714	0.351
Real-ESRGAN	0.745	0.44
Diffusion	0.692	0.357

The YOLO model makes use of convolutional layers to make character predictions. One explanation as to why the Real-ESRGAN performed better despite being less visually appealing to the human eye is that the noise in an image is much more significant quantitatively due to these analyses using pixel values and similarity. Gaussian noise in an image can significantly alter the spread of this data. Conversely, the human eye can still perceive an image under minimal Gaussian noise. Examples of this can be seen in Figure 9 and 10 in earlier sections.

A limitation is the size of the dataset used for training, state-of-the-art models such as DALLE-2 [10], Imagen [12], and Stable Diffusion [11] were trained with billions of images. General imaging data not specific towards the detection of licence plates.

7 Conclusion

The Real-ESRGAN model proved to be the better model of the two regarding SR. It is currently still the superior upscaling method of choice for LPR. However,

there are trade-offs when it comes to using each of the models. The output from the diffusion model lacks consistency, some images have a considerable amount of noise, and some images are very clear. The biggest problem with adopting diffusion models in their current state is that the iterative nature of the model means they cannot operate in real-time, making it a more viable option when technology advances in future. The real-ESRGAN images are usable but look more artificial than the diffusion ones. The Real-ESRGAN beats the diffusion model outright when evaluated with PSNR and SSIM. Although the ESRGAN outperformed the diffusion model quantitatively, further analysis of the images showed that the diffusion models were more appealing and accurate. It was noted from the results that in images at t_0 , the final output, still contained some noise. Future works could explore adding a final pre-processing step to eliminate the remaining noise, as it was evident from the results that noise affected character recognition. The fidelity of diffusion model output can be used to generate training sets. In this manner, the time it takes to generate outweighs the time it takes to collect a significant amount of data. This can be explored in future works.

References

1. Anwar, S., Khan, S., Barnes, N.: A deep journey into super-resolution: A survey. *ACM Computing Surveys (CSUR)* **53**(3), 1–34 (2020)
2. Boby, A., Brown, D.: Improving licence plate detection using generative adversarial networks. In: Iberian Conference on Pattern Recognition and Image Analysis. pp. 588–601. Springer (2022)
3. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)
4. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems* **34**, 8780–8794 (2021)
5. Diwan, T., Anirudh, G., Tembhurne, J.V.: Object detection using yolo: Challenges, architectural successors, datasets and applications. *Multimedia Tools and Applications* pp. 1–33 (2022)
6. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European conference on computer vision. pp. 184–199. Springer (2014)
7. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* **33**, 6840–6851 (2020)
8. Kim, T.G., Yun, B.J., Kim, T.H., Lee, J.Y., Park, K.H., Jeong, Y., Kim, H.D.: Recognition of vehicle license plates based on image processing. *Applied Sciences* **11**(14), 6292 (2021)
9. Lee, Y., Yun, J., Hong, Y., Lee, J., Jeon, M.: Accurate license plate recognition and super-resolution using a generative adversarial networks on traffic surveillance video. In: 2018 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia). pp. 1–4. IEEE (2018)
10. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 (2022)

11. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022)
12. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S.K.S., Ayan, B.K., Mahdavi, S.S., Lopes, R.G., et al.: Photorealistic text-to-image diffusion models with deep language understanding. arXiv preprint arXiv:2205.11487 (2022)
13. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
14. Sara, U., Akter, M., Uddin, M.S.: Image quality assessment through fsim, ssim, mse and psnr—a comparative study. Journal of Computer and Communications **7**(3), 8–18 (2019)
15. Silva, S.M., Jung, C.R.: A flexible approach for automatic license plate recognition in unconstrained scenarios. IEEE Transactions on Intelligent Transportation Systems (2021)
16. Silva, S.M., Jung, C.R.: Real-time license plate detection and recognition using deep convolutional neural networks. Journal of Visual Communication and Image Representation **71**, 102773 (2020)
17. Srivastava, A., Valkov, L., Russell, C., Gutmann, M.U., Sutton, C.: Veegan: Reducing mode collapse in gans using implicit variational learning. Advances in neural information processing systems **30** (2017)
18. Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1905–1914 (2021)
19. Wang, Z., Bovik, A.C.: Mean squared error: Love it or leave it? a new look at signal fidelity measures. IEEE signal processing magazine **26**(1), 98–117 (2009)
20. Wang, Z., Bovik, A.C., Lu, L.: Why is image quality assessment so difficult? In: 2002 IEEE International conference on acoustics, speech, and signal processing. vol. 4, pp. IV–3313. IEEE (2002)
21. Zou, Y., Zhang, Y., Yan, J., Jiang, X., Huang, T., Fan, H., Cui, Z.: License plate detection and recognition based on YOLOv3 and ILPRNET. Signal, Image and Video Processing pp. 1–8 (2021)

An IoT-Based Smart Health Monitoring System

R. Lakshmi, M. Mridula, G. Sri Gayathri,

Email.id: lakshmi.1907023@srec.ac.in, mridula.1907031@srec.ac.in,

srigayathri.1907045@srec.ac.in

Department of biomedical engineering, Sri Ramakrishna Engineering College, Coimbatore

Corresponding author: V. Srividhyasakthi, M.E, Assistant professor (Sr. G)

Email.id: srividhyasakthi.v@srec.ac.in

Department of Biomedical Engineering, Sri Ramakrishna Engineering College, Coimbatore.

ABSTRACT

As per the grandview healthcare research report published in 2022, the wearable healthcare market in India is expected to reach US\$ 372 million owing to an increase in lifestyle disorders, and better awareness of healthcare [12]. As per the world health organization (WHO) guidelines, the essential parameters to be monitored for COVID-19 patients are temperature, heart rate, SpO₂, and perfusion index [18]. Recent wearable watches such as Apple, Samsung Gear and Fitbit produce inaccurate and unreliable SpO₂ results due to movement of the skin and insufficient skin contact [20]. People affected by acute conditions as well as patients in critical care having chronic conditions require continuous monitoring of vital parameters, which results in increased production of wearables in the healthcare market. The proposed real-time vital monitor is the cost-effective, accurate, IoT-enabled, Bio Impedance Analysis [BIA] integrated, two-way communication device that is capable of monitoring as well as recording the vitals of patients, such as temperature, oxygen saturation (SpO₂), heart rate, respiratory rate, perfusion index, and hydration status. The obtained patient status is displayed in real-time using the mobile application. This is an effective and reliable way of minimizing the fatality rate and clinical admission of patients, by retaining lives in the golden hour.

KEYWORDS

Real-time vital monitoring, respiratory diseases, alert system, temperature, SpO₂, heart rate, Bioimpedance analysis.

INTRODUCTION

In accordance with a health survey conducted by the world health organization (WHO), nearly 260 million people suffered from asthma, and over 3 million deaths happened due to chronic obstructive pulmonary disorder [11]. In many hospitals, due to shortages of healthcare workers and poor monitoring facilities, vital parameters of the patients could not be continuously monitored. According to the retrospective cohort study conducted by Peru public hospital in lima concluded that oxygen saturation (SpO₂) levels below 90% act as an effective predictor for COVID-19 patients [16]. Dehydration is the third leading cause of death in infants and children [13]. Loss of 2-3% of body water content can cause physical and cognitive defects [5]. Overhydration causes lower mental ability, tachycardia, acute renal failure, and edema. The consequences of overhydration

are myocardial infarction, hypertension, edema, and seizure. Hydration status is determined by clinical symptoms which can be interpreted by doctors only.

The methods available for assessing hydration status aim to measure body fluid compartments either directly or indirectly are as follows:

1. Biomarkers assessment-which estimate the solute concentration in body fluids.
2. Neutron activation analysis-which is a non-invasive technique that uses electromagnetic radiation emission to estimate the elements of fluid.
3. Hematological indices assessment which is an invasive method that uses blood samples to estimate the hydration status.
4. Urine specific gravity test – measure hydration status by collecting urine samples and analyzing its concentration level.

The limitations of the above-mentioned methods are expensive, invasive, require biological samples, require trained personnel and time consuming.

Body fluids maintains skin elasticity. In the case of dehydration, the patient's skin loses its elasticity nature and it will take more time to regain its normal position. The skin pinch test is one of the simplest methods for accessing the dehydration state of patients. This assessment is not an effective method for measuring hydration status in serious malnutrition patients, geriatric patients and this method is also subjective.



Fig 1: Skin pinch

So, a device for monitoring hydration status is essential. To achieve this, wearable, non-invasive devices play a crucial role in long-term vital monitoring of the human body's imperative parameters in real-time. In case of any emergency situation, it alerts the caretakers as well as healthcare professionals. The internet of things technology provides global network infrastructure with quantified-self capabilities. By using IoT technology, this device provides an effective two-way communication system. The clinical decision support system aids patients with arrhythmia, respiratory diseases, dialysis, geriatric, athletes, farmers, and ICU patients.

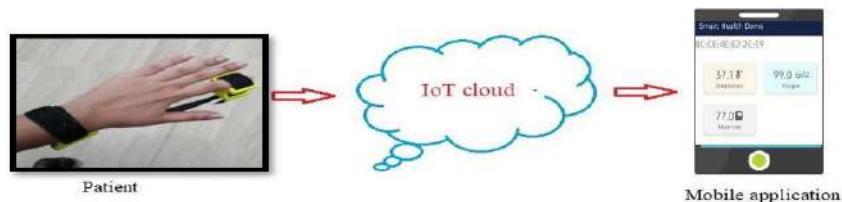


Fig 2 Architecture of smart health monitoring system

PROBLEMS WITH EXISTING SYSTEM

The current systems only measure foremost vitals such as temperature, SpO₂, heart rate. Most of the existing devices measure SpO₂ in wrist using green LED. So, it is absorbed by non-vascular tissues and provide inaccurate readings. In existing system, there is a lack of

- Two-way communication between healthcare professionals and patient.
- Alert system - To indicate users, healthcare professionals.
- Clinical Decision Support System (CDSS)
- Hydration monitoring
- Storing data in downloadable format for NABH accreditation.

COMPARISON OF PROPOSED SYSTEM WITH EXISTING PRODUCTS:

	Proposed system	Product 1 (Noise colour fit)	Product 2 (Boat flash watch)	Product 3 (Honor smart watch)	Product 4 (GOQII)
Battery	Rechargeable battery	Rechargeable battery (lithium polymer)	Rechargeable battery (lithium polymer)	Rechargeable battery(lithium ion)	Rechargeable battery (lithium polymer)
Shape	Rectangle	Rectangle	Rectangle	Rectangle	Rectangle
Width(mm)	26.5	36	36	36	36
Ambulance alert	✓	✗	✗	✗	✗
Water resistance	✓	✓	✓	✓	✓
Display size(mm)	24	38	33	36	37
Parameters to be monitored	Heart rate SpO ₂ Temperature Respiratory rate Perfusion index Hydration level	Heart rate SpO ₂ Temperature Respiratory rate Perfusion index Hydration level	Heart rate SpO ₂ Temperature Respiratory rate Perfusion index Hydration level	Heart rate SpO ₂ Temperature Respiratory rate Perfusion index Hydration level	Heart rate SpO ₂ Temperature Respiratory rate Perfusion index Hydration level
Power supply(V)	3.7	4.2	4.2	3.0 – 4.2	4.2

Table 1. Comparison of proposed solution with existing products

MATERIALS AND METHODS:

TEMPERATURE MEASUREMENT

Body temperature is one of the body's first reaction to most of the illness and infections. Temperature measurement is carried out to measure the fever spread in the human body [2]. It helps to maintain the homeostasis body condition. It is one of the key vital signs of all diseases. The requirement for temperature measurement is measurement of core body temperature which ranges from 35 °C to 40°C, high accuracy, high precision, high resolution, good sensitivity, long durability. The selected sensor has the above-mentioned requirements in order to get appropriate results.

WORKING PRINCIPLE

The working principle of selected temperature sensor is the **Resistance temperature detector (RTD)**. It will have a positive temperature coefficient i.e. As the temperature of the body increases, corresponding resistance also increases [13]. By measuring the resistance, we calculated the temperature of the patient body. This method of measuring temperature provides accurate and reliable results with a wide range of temperature. Advantage of the selected sensor is that it does not require any calibration circuit, meets standard such as ASTM E1112(American Standard for Testing and Materials) and ISO (International Standard for Organization) standards, also suitable for remote applications. By measuring that resistance, we obtained the temperature of the patient body. Based on price/performance, we have selected the sensor that meets the above requirements.

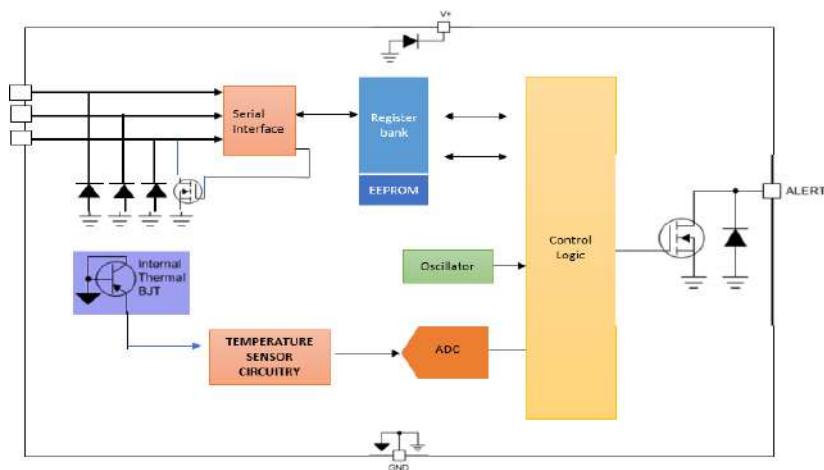


Fig 3. Circuit diagram of temperature sensor



Fig 4: Temperature sensor

Temperature sensor:

Operating temperature range	-55° to 150°C
Supply voltage	1.8 V to 5.5 V
Supply current	5 mA

Accuracy	+/- 0.1 °C
Resolution	0.0078 °C

BLOOD OXYGEN SATURATION

SpO_2 is the measure of the amount of haemoglobin saturated with oxygen as it shows the monitoring of the overall health person [2]. The SpO_2 of a patient's blood and heart rate is calculated using a SpO_2 sensor. It uses **photoplethysmography** to measure heart rate and SpO_2 . It uses infrared at 940nm (absorbed by haemoglobin) and red light at 660nm (absorbed by deoxyhaemoglobin) [17]. At the receiving end, it has a photodiode to detect how much light is absorbed. A portion of the light passes through the tissues being absorbed and remaining light strikes the photodetector. It creates a stable and non-pulsatile "DC current". Veins and capillaries become constant throughout the cardiac cycle. The variability in light absorption influences PPG data and it is measured using the PPG sensor [7].

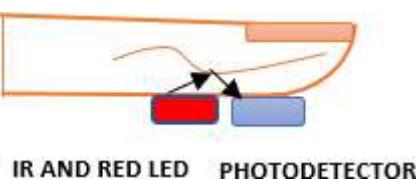


Fig 5. Photoplethysmography principle

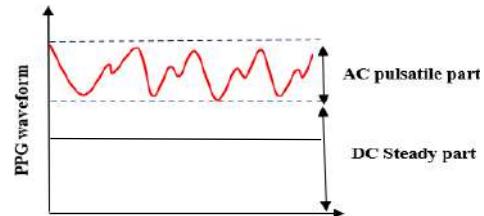


Fig 6. PPG waveform

A healthy person should have SpO_2 lying between 94% to 100% and a heart rate lying between 60 and 100 beats per minute. The requirement of the sensor is accurate measurement irrespective of skin pigmentation, skin thickness, and rejection of stray light absorption. We can achieve these features through this sensor. This sensor contains the following components such as optical elements, e.g., LEDs (Red and Infrared), photodetectors, and low noise filters with the rejection of ambient light. Benefits of the selected sensor is that it has ultra-low power heart-rate monitor (1 mW) which improves its battery life, ultra-low shutdown mode (0.7A) which permits the voltage source to remain powered at any time and it is efficient to be used in wearable devices.

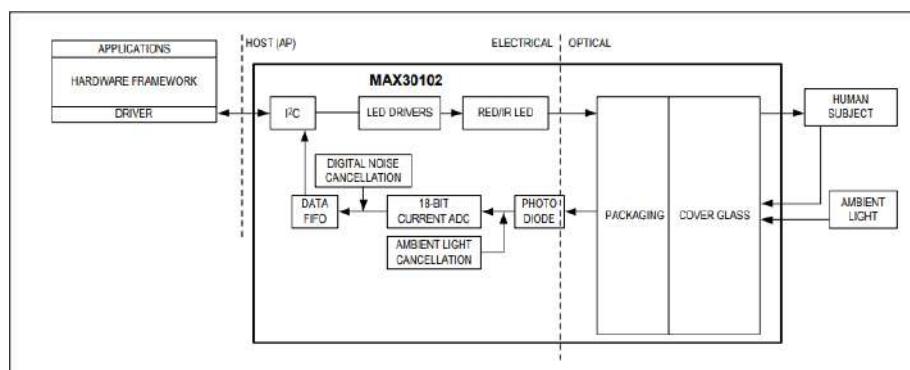


Fig 7: Circuit diagram of SpO_2 sensor



Fig 8: SpO₂ sensor

SpO₂ and Heart rate sensor:

Power supply	3.3 V
SNR ratio	High
Special features	Ultra-low shutdown current (0.7 uA), Low power rate monitor (<1 mW)

HYDRATION MEASUREMENT:

Body hydration state measurement is essential to maintain body temperature, lubrication of joints, prevention of infections, proper function of organs. The level of hydration impacted the exercise, blood thickness, body fitness activity [4]. Bioimpedance analysis is a non-invasive, objective, faster, cost-efficient, quantitative analysis method for assessment of hydration status in human body [6]. Dehydration is the third leading cause of death in infants and for 69% of geriatric patients. It causes deep vein thrombosis (DVT) [6], difficulty/failure in continuous monitoring of dialysis patients for dehydration during intra, post and inter dialytic sessions, difficulty in objective monitoring of dehydration in severe acute malnutrition patients. Bioimpedance analysis method is the simplest way of acquiring patient's hydration status by injecting high frequency, less amount of alternating current into the patient's body which adheres with the IEC60601-1 standard. Impedance value is calculated and by using this value, we can measure intracellular water (ICW), extracellular water (ECW) and total body water (TBW) accurately.

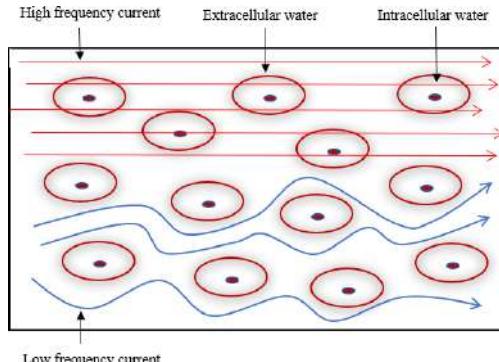


Fig 9. Passing current flow through tissues

III. Wi-fi cum Microcontroller:

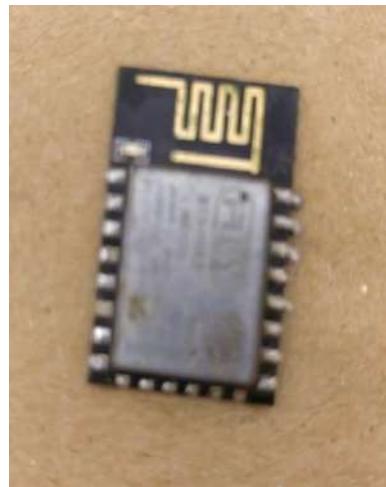


Fig 10: Wi-Fi cum microcontroller

	Tensilica 32-bit RISC CPU
Microcontroller	
Operating Voltage	3.3 V
Input voltage	7-12 V
Digital I/O pins(DIO)	16
Clock speed	80 MHz

Cloud:

Database	Real-time, Accessed from a mobile device or web browser, secure transmission of data.
-----------------	---

METHODOLOGY:

BLOCK DIAGRAM OF SMART HEALTH MONITORING SYSTEM

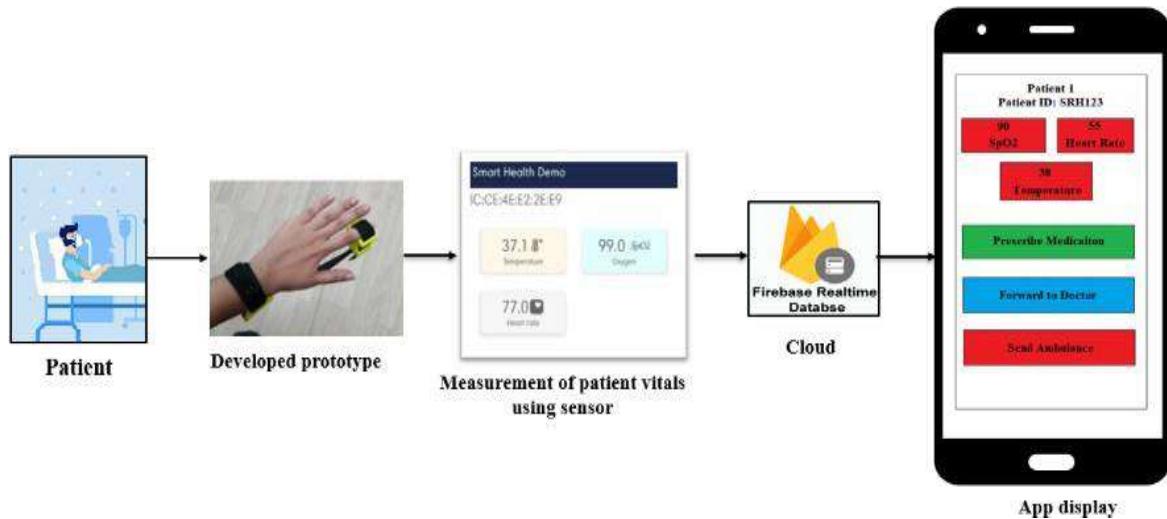


Fig 11: Block diagram of Proposed solution

FEATURES

1. Real time monitoring (Temperature, SpO₂, Heart rate, Respiration rate, Perfusion index, hydration level)
2. Non-invasive
3. Portable
4. Data security and privacy
5. Affordable
6. Ambulatory alert
7. Telemedicine
8. Two-way communication mobile application



Fig 12: Proposed solution

APP DISPLAY FOR NURSE SIDE USAGE

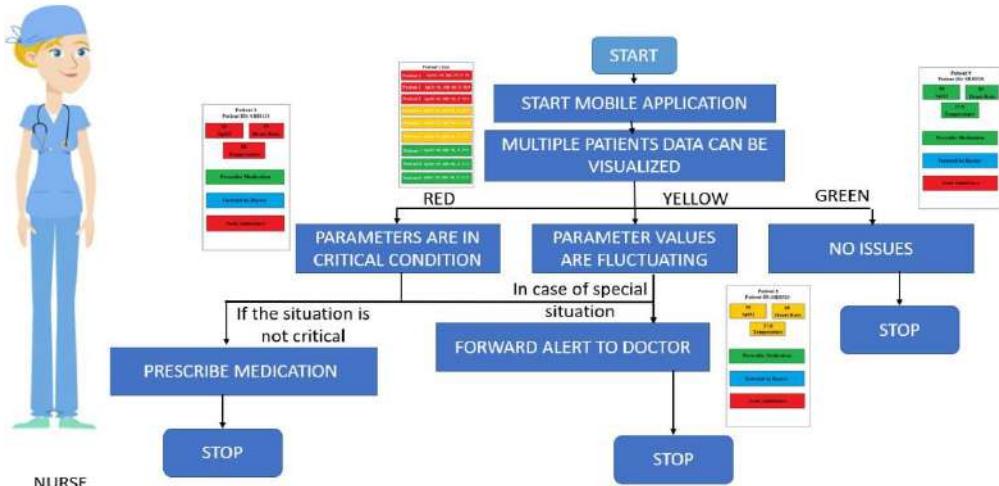


Fig 13: App display for nurse

Fig 13 depicts how the mobile application works from the nurse's point of view. Once the mobile application is opened or started the mobile application would display all the patient's vital information in a row-wise manner with colour indications, like red – for critical, yellow/orange – for vitals fluctuation, and green for normal condition. The nurse will be able to forward the alert to the doctor if the condition of the patient is in a critical condition, and the nurse would be able to alert an ambulance in case of an emergency situation.

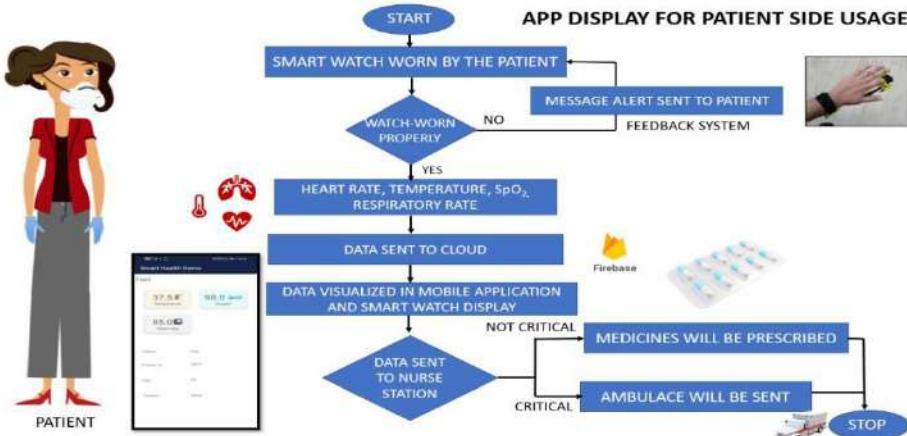


Fig 14: App display for patient

Fig 14 depicts how the mobile application works from the patient's point of view. Once the mobile application is opened or started the mobile application would display all the values of the patient's vitals that are being monitored if patient worn the smart watch properly otherwise feedback system sent the alert message to the patient. The patient will be able to view his vital values, view the medicines if prescribed, and see the ambulance location if the ambulance is being sent to the patient's location.

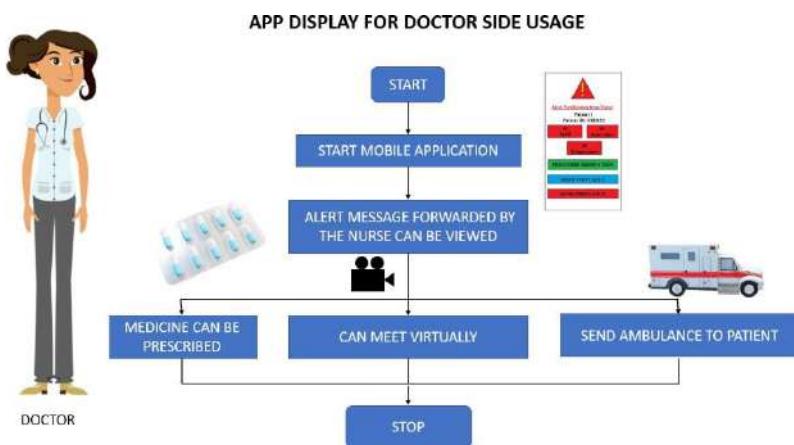


Fig 15: App display for doctor

Fig 15 depicts how the mobile application works from the doctor's point of view. Once the mobile application is opened or started the mobile application would display alert messages which is forwarded from the nurse that will appear in a row-wise manner with colour indications, like red – for critical, yellow/orange – for

vitals fluctuation. The doctor will be able to prescribe medications to a patient in a virtual mode, meet the patient virtually and send an ambulance to the patient's location in an adverse situation.

RESULT

A wearable monitoring device is designed to continuously monitor the health symptoms from remote locations. The temperature, heart rate, and SpO₂ of the patients are successfully obtained with an accuracy of 98.4%, 97%, 97.6% respectively. The hydration status is displayed with an accuracy of 98.3%. The obtained result is successfully displayed using mobile application featured with effective two-way communication system. The accuracies are validated using gold standard methods. The medical data of corresponding patients are sent to clinicians, nurses, and patients and informed to the respondents of the family to minimize the stress.



Fig 16: Comparison with gold standard method

FUTURE WORK

- Making flexible electronics
- Integrated with existing respiratory devices such as ventilator, oxygen conservative device etc.,
- Monitoring ECG by using electrodes inside wearable t-shirts

REFERENCES

1. Al Bassam, Nizar, Shaik Asif Hussain, Ammar Al Qaraghuli, Jibreal Khan, E. P. Sumesh, and Vidhya Lavanya. "IoT based wearable device to monitor the signs of quarantined remote patients of COVID-19." *Informatics in Medicine Unlocked* 24 (2021): 100588.
2. Al Bassam, Nizar, Shaik Asif Hussain, Ammar Al Qaraghuli, Jibreal Khan, E. P. Sumesh, and Vidhya Lavanya. "IoT based wearable device to monitor the signs of quarantined remote patients of COVID-19." *Informatics in Medicine Unlocked* 24 (2021): 100588.
3. Igor Didier Sabukunze; Djoko Budiyanto Setyohadi; Margaretha Sulistyoningsih et.al., "Designing An Iot Based Smart Monitoring and Emergency Alert System for Covid19 Patients" (IEEE) April 2021.
4. "A Comparative Study of Body Hydration Measurement Techniques for Health Monitoring." In 2019 6th International Conference on Computing for Sustainable Global Development (INDIACoM), pp. 1101- 1105. IEEE, 2019.
5. Khalil, Sami F., Mas S. Mohktar, and Fatimah Ibrahim. "The theory and fundamentals of bioimpedance analysis in clinical status monitoring and diagnosis of diseases." *Sensors* 14, no. 6 (2014): 10895-10928.

6. Moonen, Hanneke Pierre Franciscus Xaverius, and Arthur Raymond Hubert Van Zanten. "Bioelectric impedance analysis for body composition measurement and other potential clinical applications in critical illness." *Current Opinion in Critical Care* 27, no. 4 (2021): 344.
7. Guk, Kyeonghye, Gaon Han, Jaewoo Lim, Keunwon Jeong, Taejoon Kang, Eun-Kyung Lim, and Juyeon Jung. "Evolution of wearable devices with real-time disease monitoring for personalized healthcare." *Nanomaterials* 9, no. 6 (2019): 813.
8. Agcayazi, Talha, Gin Jong Hong, Bryan Maione, and Ethan Woodard. "Wearable infant hydration monitor." In 2016 IEEE virtual conference on applications of commercial sensors (VCACS), pp. 1-10. IEEE, 2016.
9. Munee, Amgad, Suliman Mohamed Fati, and Saddam Fuddah. "Smart health monitoring system using IoT based smart fitness mirror." *TELKOMNIKA (Telecommunication Computing Electronics and Control)* 18, no. 1 (2020): 317-331.
10. Tham, O. Y., M. A. Markom, AH Abu Bakar, ES Mohd Muslim Tan, and A. M. Markom. "IoT health monitoring device of oxygen saturation (SPO₂) and heart rate level." In 2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE), pp. 128-133. IEEE, 2020.
11. https://www.who.int/health-topics/chronic-respiratory-diseases#tab=tab_1
12. <https://www.grandviewresearch.com/industry-analysis/wearable-medical-devices-market>
13. [https://www.sciencedirect.com/topics/engineering/resistance-temperature-detector#:~:text=The%20resistance%20temperature%20detector%20\(RTD,temperature%2C%20the%20larger%20the%20resistance.](https://www.sciencedirect.com/topics/engineering/resistance-temperature-detector#:~:text=The%20resistance%20temperature%20detector%20(RTD,temperature%2C%20the%20larger%20the%20resistance.)
14. Nemcova, Andrea, Ivana Jordanova, Martin Varecka, Radovan Smisek, Lucie Marsanova, Lukas Smital, and Martin Vitek. "Monitoring of heart rate, blood oxygen saturation, and blood pressure using a smartphone." *Biomedical Signal Processing and Control* 59 (2020): 101928.
15. Whitehead, F. J., R. T. L. Couper, Linda Moore, A. J. Bourne, and Roger W. Byard. "Dehydration deaths in infants and young children." *The American journal of forensic medicine and pathology* 17, no. 1 (1996): 73-78.
16. Mejía, Fernando, Carlos Medina, Enrique Cornejo, Enrique Morello, Sergio Vásquez, Jorge Alave, Alvaro Schwalb, and Germán Málaga. "Oxygen saturation as a predictor of mortality in hospitalized adult patients with COVID-19 in a public hospital in Lima, Peru." *PloS one* 15, no. 12 (2020): e0244171.
17. Chan, Edward D., Michael M. Chan, and Mallory M. Chan. "Pulse oximetry: understanding its basic principles facilitates appreciation of its limitations." *Respiratory medicine* 107, no. 6 (2013): 789-799.
18. <https://apps.who.int/iris/bitstream/handle/10665/338882/WHO-2019-nCoV-clinical-2021.1-eng.pdf>
19. Mejía, Fernando, Carlos Medina, Enrique Cornejo, Enrique Morello, Sergio Vásquez, Jorge Alave, Alvaro Schwalb, and Germán Málaga. "Oxygen saturation as a predictor of mortality in hospitalized adult patients with COVID-19 in a public hospital in Lima, Peru." *PloS one* 15, no. 12 (2020): e0244171.
20. Phillips, Caleb, Daniyal Liaqat, Moshe Gabel, and Eyal de Lara. "WristO₂: Reliable peripheral oxygen saturation readings from wrist-worn pulse oximeters." In 2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), pp. 623-629. IEEE, 2021.

Multi stage fruit grading system*

Joel J Sebastian¹, J Krishnanunni¹, Joveal K Johnson¹, Ashik Mujeeb¹,
Anjali S¹, and Vinny Pious¹

¹Mar Baselios College of Engineering and Technology, Mar Ivanios Vidya Nagar,
Nalanchira, Thiruvananthapuram, Kerala-695015

¹joeljsebastian.official@gmail.com,krishnanunnijtvm@gmail.com,
asifmujeeb13@gmail.com,jovealk@gmail.com,anjali.s@mbcet.ac.in,
vinnypious243@gmail.com

Abstract. Fruit classification and grading is an important task in many industrial applications. Fruit grading is an important process for producers which affects the fruits quality evaluation and export market. Although the grading and sorting can be done by the human, but it is slow, labor intensive, error prone and tedious. Hence, there is a need of an intelligent fruit grading system. In recent years, researchers had developed numerous algorithms for fruit sorting using computer vision. But they have had some shortcomings like they does not use the best classification model with the best possible accuracy or that they either classify the fruit based on type or classify fruits based on ripeness. In this paper, we propose a multi-stage fruit grading system Using deep learning. This system has three phases:

1. Classification of fruits based on if it is a banana or not.
2. Classification of banana based on freshness.
3. Classification of fresh banana based on ripeness.

We have used different pretrained CNN models and identified the best ones in the different phases of the system. The experimental result shows that MobileNet, Resnet 50 and Inception V2 produce higher accuracy for phase1, phase2 and phase 3 respectively.

Keywords: Deep learning · Convolutional Neural Network · MobileNet · Resnet · Visual Geometric Group

1 Introduction

India produces 146 million tons of vegetables and 75 million tons of fresh fruits per annum. The marketing of fruits and vegetables is more challenging than many industrial products because of the perishability, seasonality, and bulkiness. Diversified consumption patterns of the Indian consumers and poor Supply Chain infrastructure makes marketing for fruits and vegetables more complicated. Effective management of the warehouses is instrumental in maintaining the supply chain.

* Supported by Mar Baselios College of Engineering and Technology.

Hence we concentrated on the warehouse part of the supply chain. A fruit classification system may be used to identify the fruit species and identify defective fruits. It may also be used to help people decide whether specific fruit species meet their dietary requirements[7],[8],[9] and [10]. India is the largest producer of bananas in the world, with a production of 297 lakh MT. Therefore we concentrated on effective ways to manage banana fruit in a warehouse. We propose a three-step approach in classifying bananas, for their efficient storage in a warehouse. In the first step, we separate bananas from other fruits. In the second step, we separate the fresh bananas from the rotten ones[4], and in the third one, we classify the bananas into four categories based on their ripeness level[2]. The different steps in our system are shown in Fig 1. Based on the literature survey that was conducted, we were able to identify that CNN-based models were able to produce better results than other proposed methods. For this project, we used some of the best-performing models to build an effective system to classify bananas. Warehouse management can use this information in organizing and managing related warehouse activities.

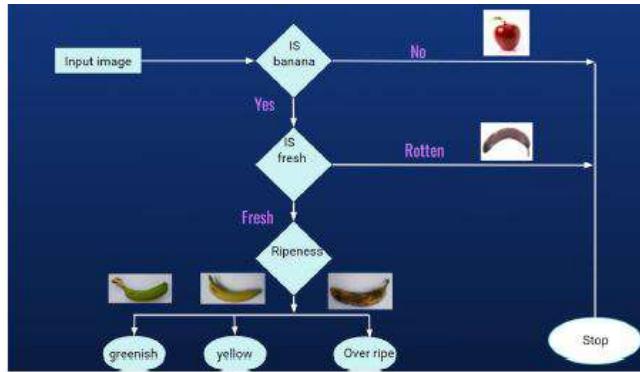


Fig. 1. Architecture Diagram

2 Related Works

D. Karakaya, O. Ulucan and M. Turkan[3] conducted a study to comparatively analyse an image dataset containing samples of three types of fruits to distinguish fresh samples from those of rotten. They also proposed a vision based framework which utilizes histograms, gray level co-occurrence matrices, bag of features and convolutional neural networks for feature extraction. The classification process is carried out through well known support vector machines based classifiers. After testing several experimental scenarios including binary and multi-class classification problems, it turned out that highest success rates

are obtained consistently with the adoption of the convolutional neural networks based features.

S. Chakraborty, F. M. J. M. Shamrat, M. M. Billah, M. A. Jubair, M. Alauddin and R. Ranjan[4] used 3 deep learning methods: max pooling, average pooling and mobile net, to classify fruits based on whether they are rotten or fresh. Mobile net had the best accuracy with 99.06 percent while max pooling and average pooling had an accuracy of 94.49 and 93.06 percent respectively.

Amruta Supekar, Madhuri Wakode[5] classified mangoes into different grades based on different parameters such as colour, size and shape. They used a pre-trained random forest classifier for this purpose which gave an accuracy of 99.5 percent.

3 Proposed Work

3.1 Problem Statement

To develop, compare and analyse different Deep Learning models for identifying if the fruit is a banana, the quality of the banana and also grading it into different ripening levels.

3.2 Methodology

Automatic classification of fruit freshness play an important role in the food industry. But the traditional methods that detect the freshness of fruits are slow, laborious and time consuming. Therefore, we propose an approach for fruit freshness classification which can potentially reduce human efforts. This can also reduce the cost by identifying the defects in the fruits. In this approach we classify banana based on their freshness level and also grade them based on their ripening levels.

Level 1 The pretrained MobileNet models were used in phase 1. Mobilenet can classify about 1000 different objects. Here V2, V3 are improvements over the version 1. Before feeding the image of object they are first preprocessed. Preprocessing includes resizing to target size 224 * 224, then they are converted to an array and then expanded to fit the model. The image pixel values are mapped from (0,255) to (-1,1).The image is classified as banana or not based on the highest value from the result array.

Level 2 After comparing various models such as VGG-16,mobilenet V2 V3 ,Inception V2 and Resnet 50 Resnet 50 V2, we found out that Resnet 50 gives the best performance.

Level 3 After comparing different models the best one was found out to be Inception V2.

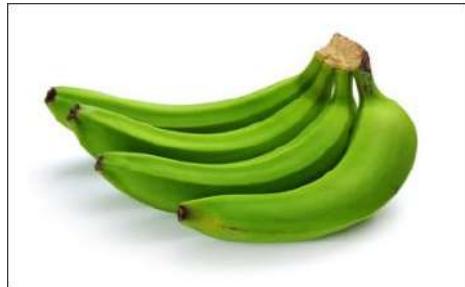


Fig. 2. Input Image
[[('n07753592', 'banana', 0.972868)]]

Fig. 3. Result of level 1 classification

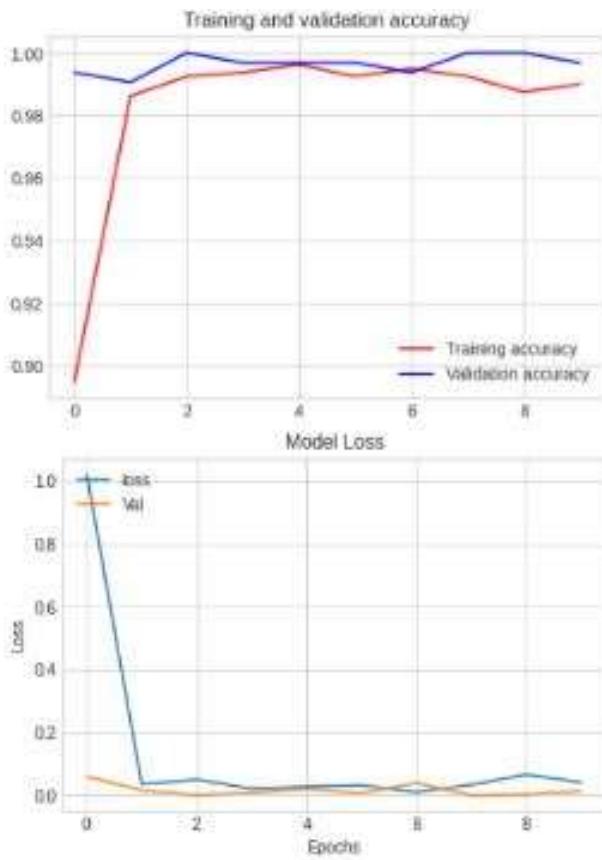


Fig. 4. Accuracy and loss of Resnet 50

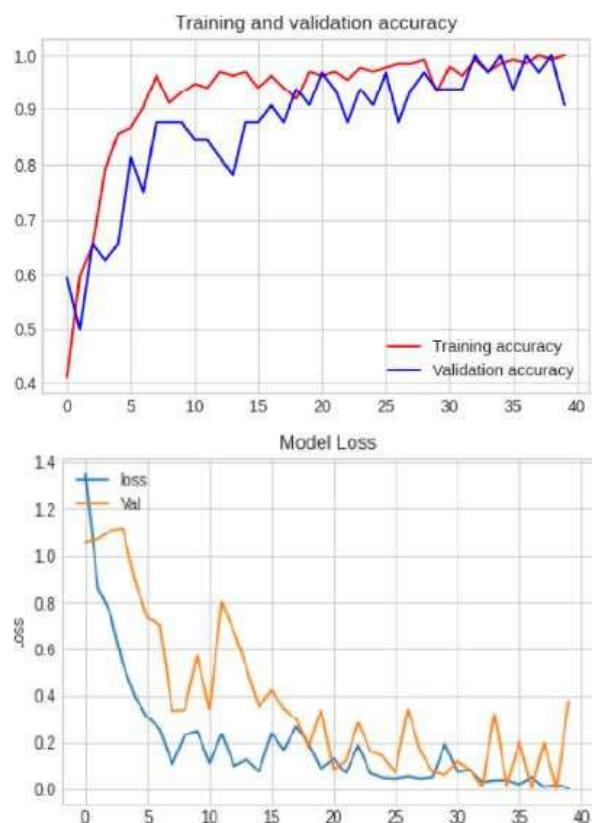


Fig. 5. Accuracy and loss of Inception V2

4 Result analysis

After trying out aforementioned models by varying the epochs, batch sizes and optimizers the following conclusions can be made:

At stage 1, for identifying whether the fruit is banana or not, both mobilenet V1 and mobilenet V2 have shown the best performance. At stage 2, for classifying rotten bananas from fresh ones Resnet50 and Resnet50 V2 have proven to be the best with accuracy 100 percent at epoch 20 with adam as optimizer. At stage 3, for classifying fresh bananas to green, yellow and overripen, Inception V2 has come out as the best model with accuracy of 99.8 percent with adam as optimizer.



Fig. 6. Best models

Performance of various models are as follows:

4.1 Based on number of epochs

Model	Accuracy
Mobilenet V1	97%
Mobilenet V2	97%
Mobilenet V3	93%

Fig. 7. Level 1

Model	Epoch-10		Epoch-20	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
Mobilenet V2	99.79%	100%	100%	100%
Resnet 50	100%	99.93%	100%	100%
Resnet 50 V2	100%	99.89%	100%	100%
Inception V2	99.90%	99.87%	100%	99.87%
Inception V3	99.83%	100%	100%	100%
VGG	99.68%	99.37%	99.84%	99.69%

Fig. 8. Level 2

Model	Epoch-10		Epoch-20	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
MobilentV2	99.79%	46%	99%	18%
Resnet 50	95%	84%	96%	96%
Resnet 50 V2	93%	66%	96%	96%
Inception V2	99.90%	99.87%	100%	99.87%
Inception V3	97%	82%	95%	87%

Fig. 9. Level 3

4.2 Based on batch size

Model	Batches	
	batch 16	batch 32
MobilentV2	100%	98%
Resnet 50	100%	99%
Resnet 50 V2	99%	99%
Inception V2	100%	100%
Inception V3	100%	99%
VGG16	98%	98%

Fig. 10. Level 2

Model	Batches	
	batch 8	batch 16
MobilenetV2	81%	81%
Resnet 50	96%	84%
Resnet 50 V2	96%	59%
Inception V2	96%	95%
Inception V3	97%	96%

Fig. 11. Level 3

4.3 Based on optimizer used

Model	Adam		Adagrad	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
Mobilenet V2	99.79%	100%	99%	98%
Resnet 50	100%	100%	100%	98%
Resnet 50 V2	100%	100%	97%	99%
Inception V2	99.90%	99.87%	100%	99.87%
Inception V3	99.83%	100%	99%	99%
VGG16	99.68%	99.37%	99%	97%

Fig. 12. Level 2

Model	Adam		Adagrad	
	Train accuracy	Test accuracy	Train accuracy	Test accuracy
MobilenetV2	99.79%	46%	98%	62%
Resnet 50	96%	96%	88%	93%
Resnet 50 V2	96%	96%	73%	87%
Inception V2	100%	99.87%	93%	83%
Inception V3	82%	97%	84%	95%

Fig. 13. Level 3

4.4 Conclusion and future scope

A three-phase architecture based on computer vision and various deep learning methods such as Inception V2, Inception V3, Resnet50, Resnet50 V2, MobileNet, MobileNet V3, VGG-16 and MobileNetV2 for fruit classification, freshness checking and ripeness classification was proposed in this paper. We came to the conclusion that some of the model showed better results than others such as MobileNet for checking whether the fruit was a banana or not, Resnet 50 for classifying fruit based on freshness and Inception V2 to classify the fruit based on ripeness. Excellent accuracy was noted during all three phases.

As for future work, a fourth phase will be added to the framework which will classify the fruit based on the mode of ripening—whether the fruit was artificially ripened using chemicals or not[6].

This method of three-phased approach is targeted at a warehouse facility in the hopes that it brings about a deeper insight to the future researchers and entrepreneurs who wants to engage in the thriving business of banana market and export.

References

1. M. Shamim Hossain, Muneer Al-Hammadi and Ghulam Muhammad *Automatic Fruit Classification Using Deep Learning for Industrial Applications*, IEEE transactions on Industrial Informatics, Vol. 15, No. 2, February 2019.
2. Mazen, F.M.A., Nashat, A.A. *Ripeness Classification of Bananas Using an Artificial Neural Network*, Arab J Sci Eng 44, 6901–6910 (2019). <https://doi.org/10.1007/s13369-018-03695-5>
3. D. Karakaya, O. Ulucan and M. Turkan *A Comparative Analysis on Fruit Freshness Classification*, 2019 Innov. Intell. Syst. Appl. Conf., pp. 1-4, 2020.
4. S. Chakraborty, F. M. J. M. Shamrat, M. M. Billah, M. A. Jubair, M. Alauddin and R. Ranjan *Implementation of Deep Learning Methods to Identify Rotten Fruits*, 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1207-1212, 2021.
5. Amruta Supekar, Madhuri Wakode *multi-parameter based mango grading using Image Processing and Machine Learning*, INFOCOMP, v. 19 no. 2, p. 175-187, December 2020
6. Harshad Vaviya, Ajaykumar Yadav, Vijaykumar Vishwakarma and Nasim Shah *Identification of Artificially Ripened Fruits Using Machine Learning*, 2nd International Conference on Advances in Science and Technology (ICAST), April 2019.
7. Hongshe Dang, Jinguo Song and Qin Guo *A Fruit Size Detecting and Grading System Based on Image Processing*, 2010 Second International Conference on Intelligent Human-Machine Systems and Cybernetics.
8. Vibha V, Vinamra and Sampada K S *Fruit Grading to Assist Selection of Fresh Fruits*, 2021 International Conference on Circuits, Controls and Communications (CCUBE).
9. Huai-Kuei Wu, Jui-Sheng Wang and Yuan-Hsin Chen *Development of Fruit Grading System Based on Image Recognition*, 2020 IEEE 2nd International Conference on Architecture, Construction, Environment and Hydraulics (ICACEH).

10. M. Shamim Hossain, Muneer Al-Hammadi and Ghulam Muhammad *Automatic Fruit Classification Using Deep Learning for Industrial Applications*, IEEE Transactions on Industrial Informatics (Volume: 15, Issue: 2, February 2019).

Sentiment Analysis on feedback data of E-commerce products based on NLP

* Dr.K.Sumathi¹, Dr.Kundhavai Santharam²

¹Assistant Professor, Department of BCA,
The American College(autonomous), Madurai
Sumathirajkumar2006@gmail.com

² Associate Professor, Department of Business Administration, Kalasalingam
Academy of Research and Education, Krishnan Koil
skundhavair@gmail.com

Abstract. In today's competitive business world, companies seek strategies to keep their existing customer base satisfied while also attracting new customers by launching new offers or products with a mix of chosen marketing mix. It has become necessary to keep track of whether customers like or dislike a product and how they react to a particular offering in order to improve the service and stay ahead of competitors. The focus of this paper will be on using NLP technique- sentiment analysis, which analyses the customer sentiments based on customer reviews / feedback data for a product, which benefits businesses by keeping track of consolidated review on a product. In this paper, we used Amazon product review datasets from kaggle, data. world and performed sentiment analysis to understand the overall customer sentiments based on customer reviews in a better and refined manner. The current study's dataset includes 30653 customer reviews for various products. Python NLTK Library has been used for pre-processing and to generate a model using the Regression Algorithm. The model performs well with F1-Score .87.

Keywords: Natural Language Processing for Sentiment Analysis, Analysis of Product Reviews, Customer feedback Analysis

1. INTRODUCTION

Today the evidence of Natural Language Processing(NLP)'s cutting-edge impact is realized by many decision makers in corporate and other sectors. In every wink of a moment each and everyone seeks to turn towards resolving issues at one point of time or the other in their field. Every individual is unique by their own ways and their opinions vary from one person to another in varied circumstances. There arises a need to search for a process that enables one to help categorize the opinions of people. The process of computationally identifying opinions and categorizing can be expressed in a piece of text, particularly in order to determine whether the writer's attitude toward a particular topic, product, etc. is positive, negative, or neutral, is known as sentiment analysis. Sentiment analysis, also known as opinion mining, is a natural language processing (NLP) technique that identifies the emotional tone of a body of text. This is a popular method for

businesses to determine and categorize customer opinions about a product, service, or idea. Sentiment analysis is a powerful marketing tool that allows product managers to understand customer emotions and incorporate them into marketing campaigns. It is an important factor in product and brand recognition, customer loyalty, customer satisfaction, advertising and promotion success, and product acceptance. Any corporate company would always look out to choose a benchmarking strategy to acquire its customers and in this study the opinions of the customers recorded in social media platform are taken into consideration.

This is an era with a buzz word 'online shopping' and writing instant reviews of their shopping experience. The reviews and ratings play a vital role for a product life cycle in the market irrespective of its sector. The ratings and reviews are taking the form of emotions being expressed by the customers. On the basis of the review the product manufacturers predict their likely sales alternating their marketing mix. In the current study the researchers have identified a solution to measure the emotions / opinions which is recorded online through the forms of reviews / a set of texts. In the view of extracting information for sentiment analysis, a Machine Learning (ML) and Natural Language Processing (NLP) engine is used in the study. In general, machine learning enables the software to improve its ability to predict the results of analyses without being explicitly coded. In essence, it gives the Programme the ability to "learn" from the experience and develop over time. NLP is used to examine human language and decipher its meaning. This includes terminology extraction, grammatical analysis and text segmentation. While we analyze the suitable algorithmic technique for the sentiment analysis, the ML and NLP is used as tools to derive the final inferences and findings. Three categories of algorithms are typically used namely – Role - based, Automatic and Hybrid respectively.

- **Rule-based** - This is the simplest and most straightforward strategy to use. It evolves manual predictions with set of rules created, which aid the system in analyzing the content it reads. The limitation of relying on manual inputs is that when the numbers increases and evolves in lots, it becomes difficult unlike an automated one.

Automatic - This is the most sophisticated method, combining ML and NLP. Thousands of phrases that have already been classified as negative, neutral or positive are initially fed into the system which is the phase known as "training." Then, armed with its newly acquired information, it may go to the next stage - "prediction", comprehend brand-new terminologies, and properly categorize

them. But the drawback which is inevitable is that the algorithm tends to commit errors, and it might often be a challenging and herculean task to determine the reason underlying the committing of errors.

- **Hybrid** - This algorithm combines the greatest features of both worlds. This method quickly runs through new phrases and expressions while utilizing the rule-based algorithm's at a high level of accuracy.

The sentiment analysis tool can review the unending content and rate it based on its negative, neutral, or positive sentiment and the decision makers would be thankful for its existence as a sophisticated algorithm in place.

2. REVIEW OF LITERATURE

While we investigate the related work it is noted that in recent past various researchers in the field of information technology and marketing are keen in investigating into the opinions recorded by the customers and sentiment analysis / opinion mining is utilized widely. Şükrü Ozan All Authors (2018) attempts to resolve the problem that is faced by the company in terms of data segmentation using various machine learning methods.

Bhatnagar, Anisha.&Bhatia, Madhulika., (2022), in their study has evolved the sentiment analysis and has highlighted that it not only enables one to segregate the information from the demographic perspective but also from the subjective perspective of handling the information as well. The process highlighted is that the use of Spectral clustering and extraction through cluster labels in order to perform supervisory based classification with the help of supervised algorithms. The accuracy level has also been checked in their study. In our current study the model is being run to check the accuracy level and the findings and conclusions are arrived at facilitating effective decision making.

While citing the work by Erik Cambria., Björn Schuller., Yunqing Xia & Catherine Havasi., (2013) it is noted that 'Both fields use data mining and natural language processing (NLP) techniques to discover, retrieve, and distill information and opinions from the World Wide Web's vast textual information'. The also indicates that it is critical and have mentioned the task of analyzing the textual information available on worldwide web to be a challenging one. Especially, the aspects such as explicit, implicit, regular, irregular, syntactical and semantic language rules is very difficult and one have to be meticulous. In the current study the researchers have meticulously taken into consideration the above aspects while analyzing the textual information.

Madan M. Batra., (2017) in Competition forum has published a work related to CX – which is Customer Experience. The customer experience is witnessed with the interaction(s) one makes with that of the company. The interactions may be either through oral reviews, printed ones, textual information shared or recorded in an open forum, etc. In their study it is indicated that the customer experience based study shall help both the companies by paving a way to achieve customer service excellence and academicians by way of formulating a research agenda about the emerging trend or forecasts related to customer experience topics. In the paper the challenging aspects of the companies such as calculating the return on investment with the customer experience is highlighted. Hence, the opinion mining and sentiment analysis helps the companies to convert the findings and inferences in terms of monetary value and calculate the return on investment.

Another study conducted by Zhenning Xu., Colin Vail., Amarpreet S. Kohli & Saeed Tajdini., (2021) has utilized word clouds, and cluster and word association analyses for understanding the unstructured textual data available on brand-owned social media. The current study also with the sentiment analysis has taken into consideration the datasets available through Kaggle – the machine learning and data science community. The paper indicates that the customer sentiments are expressed at varied levels of mindsets of customers based on the perceived brand's core position. Hence, it is clearly indicative that the customer sentiments take different stance from varied experiences, which makes the analysis task to be a challenging one.

Simkin, L., (2013) in the report compiled reflects that the analytics has gone a long way where from the level of directors, company leadership teams are used to getting updates through the market intelligence and other advanced methods of customer updates. Therefore the sentiment analysis the prime step towards the level of advanced methods of market intelligence and getting customer updates. It has also tapped upon the aspect of CRM systems that harnesses the associated customer data.

Nurfadhlina Mohd Sharef., Harnani Mat Zin and Samaneh Nadali., (2016) has shared insights based on the big data availability being vast due to decreased cost in data storage and computing power especially and previously we were in an era of depending on the transactional data wherein with the current scenario – evolves big data and ensures delivery of business value. Hence, the current study also validates the point that add to business value to the current setup when these sentiment analysis is performed and taken towards the step of big data analysis and analytics based initiatives for the companies.

The data from social media platform – twitter has been taken into consideration for using sentiment analysis elaborated in a conference paper by Md. Rakibul Hasan; Maisha Maliha; M. Arifuzzaman (2019). They have also highlighted that the analysis helps the business entity to have an effective product marketing. In their study they have incorporated bag of words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF) model concept to analyze sentiment. In the current study the textual information has been extracted from machine learning and data science platforms that are available online and as an open source. The use of Term Frequency-Inverse Document Frequency (TF-IDF) model concept to analyze sentiment has also been evolved and the accuracy has been improved with the use of the model.

Analyzing in general a qualitative data is a crucial task, in recent past NLP is used to analyze qualitative data. Kevin Crowston ,Eileen E. Allen &Robert Heckman., (2011) in their paper had initiated to take the human natural language processing and tried to automate content analysis by means of extracting theoretical evidence in an effective manner. The paper is an effective contribution as it is the first to attempt the data which is qualitative in nature. Hence, with this concept of taking into consideration the qualitative data as the basis of the study, the current study also has taken the qualitative data that is recorded in the forms of set of texts on the social media reviews.

Enshuo Hsu et al (2022),, in their work has taken the scanned documents of electronic health records which has been a challenge that is faced by many and analyzed using NLP and other image pre-processing methods. Almost taking 7 machine learning models into consideration, the study has initiated the analysis. Hence, in our study, the pre-processing and other steps are followed to ensure the accuracy level. Indrawati; Andry Alamsyah.,(2017) in their conference paper has highlighted about the data analytics use for market segmentation in telecommunication industry in Indonesia and they had called their analysis work to be social network data analytics based community detection method. The segregated groups and their views and responses or topics are analyzed and the attitude towards the product is found out.

3. PROPOSED WORK

The process of performing sentiment analysis on a dataset containing product reviews and data labels such as "Positive," "Neutral," and "Negative" sentiment for these reviews consists of several steps, such as i) data collection ii) pre-processing using NLP Techniques iii) selecting the appropriate algorithm for classification and iv) determining the metrics of analytical model respectively.

The data flow diagram of Sentiment analysis on customer reviews is given in Figure 1.

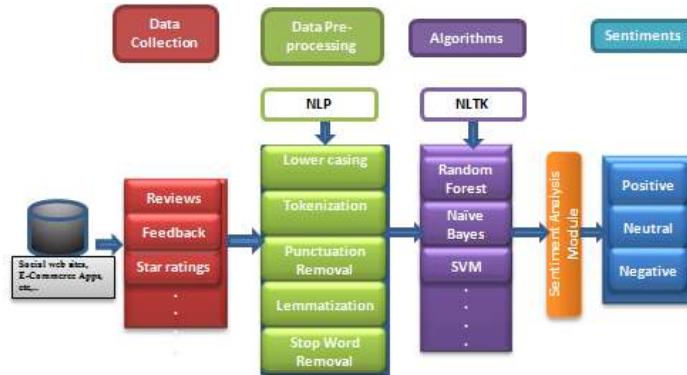


Figure 1: Sentiment analysis on customer reviews –Workflow

3.1 Data Collection

The dataset, which includes various product reviews for sentiment analysis, was obtained from kaggle and data.world. The obtained dataset contains 30653 instances, each with 20 attributes that include product and customer details in various aspects. The preview of the dataset is given in Figure 2.

```
df=pd.read_excel("Customer_Prod_Reviews.xlsx",usecols=['Brand Name','Sentiment','Reviews'])
df[0:]
```

	Brand Name	Sentiment	Reviews
0	Kindle Paperwhite	positive	Paperwhite voyage, no regrets!
1	Kindle Paperwhite	positive	One Simply Could Not Ask For More
2	Kindle Paperwhite	positive	Great for those that just want an e-reader
3	Kindle Paperwhite	positive	Love / Hate relationship
4	Kindle Paperwhite	positive	I LOVE IT
...
30648	Amazon	positive	Xmas gift
30649	Amazon	positive	yes it is a great tablet.
30650	Amazon	positive	You get a lot for the price!
30651	Amazon	positive	You get the entire World for less than \$100!
30652	Amazon	positive	You get what your paying for

30653 rows × 3 columns

Figure 2: Data set - Preview

Only three relevant features were extracted and saved in a separate Excel file for analysis by the researchers: "Brand Name", "Sentiment" and "Reviews." This file has been uploaded to Google Drive so that it can be

easily accessed from Google Colab. Kalpak K.Kulkarni et al (2020) attempted to analyse viral advertising and demonstrate that sentiment analysis is a promising tool for quantifying customer responses. The text data was pre-processed using NLP techniques and the samples were classified using the Random Forest Classifier. The procedure for performing sentiment analysis is as follows: Pre-processing of text, Converting Text to Vectors, and Classification.

3.2 Pre-processing of text

To remove extraneous information from the text, which may result in a less accurate model, data pre-processing is required. Lower casing, Tokenization, Special Characters Removal, Lemmatization, and Stop Word Removal must all be performed during the text preprocessing phase. Virmani.C., Pillai.A., Juneja.D., (2017) has conducted a study with the help of extracted information from social media using NLP and they have postulated mainly on the aspect that NLP is evolved to enhance the accuracy of visualizing the structured data available. Therefore in our study to the data pre-processing has been evolved.

a) Lower casing

Lower casing is a common text pre-processing step that is applied based on the problem statement. When the dataset contains text that is used to recognize people's emotions, this step can be skipped because doing so will result in information loss, such as using upper case words to show anger or excitement. Reviews after lower casing is displayed in Figure 3.

```
# Convert the titles to lowercase
df['review_p'] = df['Reviews'].map(lambda x: x.lower())
df[['Reviews','review_p']].head(5)
```

	Reviews	review_p
0	Paperwhite voyage, no regrets!	paperwhite voyage, no regrets!
1	One Simply Could Not Ask For More	one simply could not ask for more
2	Great for those that just want an e-reader	great for those that just want an e-reader
3	Love / Hate relationship	love / hate relationship
4	I LOVE IT	i love it

Figure 3: Lower Casing

b) Tokenization

Tokenization is the next step in pre-processing text data. It involves breaking each sentence down into individual words. Tokenization assists in interpreting the meaning of the text by analysing the word sequence. However, it is a common method for analysing large amounts of text data. It is efficient and convenient for

computers to analyse text data by examining what words appear in an article and how many times these words appear, and it is sufficient to provide insightful results. The output of Tokenization process is shown in Figure 4.

	review_p	review_p1
0	paperwhite voyage, no regrets!	[paperwhite, voyage, , , no, regrets, !]
1	one simply could not ask for more	[one, simply, could, not, ask, for, more]
2	great for those that just want an e-reader	[great, for, those, that, just, want, an, e-re...]
3	love / hate relationship	[love, /, hate, relationship]
4	i love it	[i, love, it]

Figure 4: Tokenization

c) Special Characters Removal

Special characters such as brackets, commas, and so on add no value to the text information and can be removed. The built-in library in Python named “string” which includes an attribute string.punctuation with a predefined list of punctuations that can be used to remove punctuation. Word tokens before and after Punctuation removal is shown in Figure 5.

	review_p1	review_p2
0	[paperwhite, voyage, , , no, regrets, !]	[paperwhite, voyage, no, regrets]
1	[one, simply, could, not, ask, for, more]	[one, simply, could, not, ask, for, more]
2	[great, for, those, that, just, want, an, e-re...]	[great, for, those, that, just, want, an, e-re...]
3	[love, /, hate, relationship]	[love, hate, relationship]
4	[i, love, it]	[i, love, it]

Figure 5: Punctuation Removal

d) Lemmatization

Lemmatization necessitates the word's part of speech tag value within the sentence. The lemmatization with NLTK will be ineffective unless you know whether the word is used as a verb, noun, or adjective. To perform lemmatization effectively with NLTK, use the "wordnet" and "pos" parameters. We have used WordNetLemmatizer.lemmatize. the output of lemmatization is given in Figure 6.

```

df.loc[:, 'review_p3'] = df['review_p2'].apply(lemma)
df[['review_p2', 'review_p3']].head(5)

[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!

      review_p2                  review_p3
0  [paperwhite, voyage, no, regrets]  [paperwhite, voyage, no, regret]
1  [one, simply, could, not, ask, for, more]  [one, simply, could, not, ask, for, more]
2  [great, for, those, that, just, want, an, e-re...]  [great, for, those, that, just, want, an, e-re...]
3  [love, hate, relationship]  [love, hate, relationship]
4  [i, love, it]  [i, love, it]

```

Figure 6: Lemmatization

e) Stop Word Removal

Stop word removal is supported by NLTK, and the list of stop words is available in the corpus module. To remove stop words from a sentence, divide it into words and then remove the word if it appears in NLTK's list of stop words. A stop word is a commonly used word such as "the," "a," "an," or "in" that a search engine has been programmed to ignore, both when indexing entries for searching and retrieving them as the result of a search query. These words are not required to take up valuable database space or processing time. These words can be easily removed by keeping a list of words that you consider stop words. Python's NLTK (Natural Language Toolkit) has a list of stop words stored in 16 different languages. They are located in the NLTK data directory. Product reviews after removing stopwords are shown below in Figure 7.

```

df.loc[:, 'review_p4'] = df['review_p3'].apply(remove_stop_words)
df[['review_p3', 'review_p4']].head(5)

      review_p3                  review_p4
0  [paperwhite, voyage, no, regret]  [paperwhite, voyage, regret]
1  [one, simply, could, not, ask, for, more]  [one, simply, could, ask, more]
2  [great, for, those, that, just, want, an, e-reader]  [great, those, just, want, e-reader]
3  [love, hate, relationship]  [love, hate, relationship]
4  [i, love, it]  [love]

```

Figure 7: Stop word Removal

3.3. Converting Text to vectors

Basically the raw text is not directly understood by ML models. As a result, text data must be converted into vectors that encode the contextual meaning of words and keep similar words together in dimensional space. Words like phone, mobile,

and cell phone, for example, will appear closer to each other in dimensional space and further away from unrelated words like 'nice,' 'good,' and so on. The process of converting text into vectors is known as word embedding, and the Word2Vec algorithm is used to generate these embeddings shown in Figure 8.

Word2Vec employs a neural network with a single hidden layer to predict the likelihood of each word in a text corpus being closer to the input word in dimensional space. The intuition here is that similar words are used in similar contexts and appear closer to each other in dimensional space, so the probability for similar words should be higher, but the goal of training a neural network in the Word2Vec algorithm is to learn weights of the hidden layer rather than using resulting neural network itself. These weights are our word vectors, which provide us with the location of words in a three-dimensional space. Gensim is an open-source Python natural language processing (NLP) library that provides Word2Vec model algorithms for learning word associations from a large corpus of text.

	0	1	2	3	4	5	6	7	8	9	...	90	91	92	93	94	95	96	97
0	0.056615	0.065734	0.090617	0.029363	-0.074472	0.037963	-0.073882	-0.093043	0.081691	0.077562	...	0.109322	-0.021679	0.018716	0.027730	0.213305	-0.026812	0.119108	-0.094086
1	0.051216	0.060802	0.086042	0.028500	-0.064490	0.035833	-0.066353	-0.084697	0.071118	0.070390	...	0.098208	-0.023902	0.017924	0.023872	0.193621	-0.025170	0.108167	-0.079240
2	0.049157	0.065637	0.090771	0.030776	-0.062989	0.037733	-0.071435	-0.087378	0.073743	0.071277	...	0.103724	-0.022815	0.017004	0.026823	0.194423	-0.024438	0.112997	-0.078949
3	0.047635	0.061677	0.088823	0.028528	-0.065832	0.037906	-0.072071	-0.083881	0.074879	0.076037	...	0.103268	-0.025031	0.015790	0.025511	0.192823	-0.027112	0.111052	-0.082467
4	0.055679	0.063380	0.089261	0.028232	-0.067838	0.040196	-0.071009	-0.090564	0.079632	0.075572	...	0.099588	-0.022877	0.012711	0.024774	0.207597	-0.023362	0.111581	-0.090164

5 rows × 100 columns

Figure 8: Word Embedding

3.4. Classification

Now that we have vectors for each word in our corpus, we will take the average of all the vectors in a sentence, resulting in an average array of 100 dimensions for that review statement. The features of our statement will be represented by this 100-dimension average array, which will be used by the classification model to predict sentiments and used by the classification model to categorize the sentiments.

Dataset is split into training and test dataset. Percentage split used for this analysis is 20. Training dataset is used to train the model where as test dataset is used to evaluate the performance of the model. Classification model has been trained on the data to predict sentiments once we have the features of reviews.

```

import time

#Import the DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

#Initialize the model
clf_decision_word2vec = RandomForestClassifier(
n_estimators= 80,
min_samples_split= 2,
min_samples_leaf= 2,
max_features= 'auto',
max_depth= 12,
criterion= 'entropy',
bootstrap= True,
class_weight='balanced')

start_time = time.time()

# Fit the model
clf_decision_word2vec.fit(word2vec_df, Y_train['sentiment'])
print("Time taken to fit the model with word2vec vectors: " + str(time.time() - start_time))

Time taken to fit the model with word2vec vectors: 7.661540746688843

```

Figure 9: Model Training

4. EVALUATION OF TRAINED MODEL

In order to evaluate the model's performance, a classification report and confusion matrix was created. For the text classification problem, the Gensim word2vec embedding method and Random Forest classification model produced high accuracy. Random Forest was selected for this problem statement because it outperforms other classification algorithms (Support Vector Machines, Logistic Regression) across all evaluation metrics, including Precision, Recall, and F1.

Classification Report – Training Dataset

```

train_predictions = clf_decision_word2vec.predict(train_features)
print(classification_report(Y_train['Sentiment'],train_predictions))

precision    recall   f1-score   support
negative      0.36      0.97      0.52     1346
neutral       0.55      0.82      0.66     1884
positive      1.00      0.87      0.93    22091

accuracy         --        --        --    24521
macro avg       0.64      0.89      0.71    24521
weighted avg    0.94      0.87      0.90    24521

```

Figure 10: Classification Report – Training Dataset

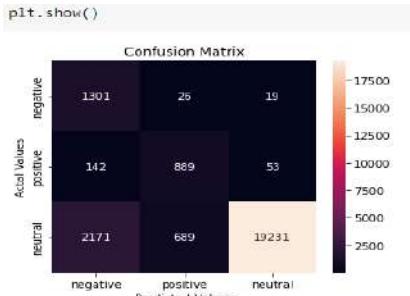


Figure 11: Confusion Matrix – Training Dataset

The number of instances used for analysis is 30653, and the split between train and test data is 80%. The model is trained on 80% of the dataset; approximately 24,521 instances are used to train the model, and approximately 6131 instances are used to evaluate the model's performance. Once the model has been trained, its performance is evaluated using the same training dataset. Figure 10 shows the F1-

score of the model for the training dataset, and Figure 11 shows the model's confusion matrix.

Classification Report – Test Dataset

The performance of the same model is evaluated using the test dataset, which contains the remaining 6131 instances of the obtained dataset. Figure 12 shows the model's F1-score for the test dataset, which is 86%, and Figure 13 shows the model's confusion matrix for the test dataset.

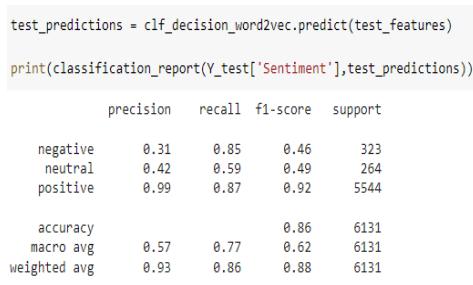


Figure 12 : Classification Report – Test Dataset

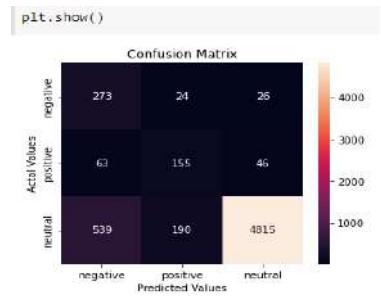


Figure 13: Confusion Matrix – Test Dataset

The total number of Amazon products available in the test dataset is 3202. 209 reviews are neutral, indicating that customers are satisfied with the product, 2755 reviews are positive, indicating that customers like the product, and 238 reviews are negative, indicating that customers dislike the product. We can accomplish this by creating a word cloud of positive feedback that is shown in Figure 15.

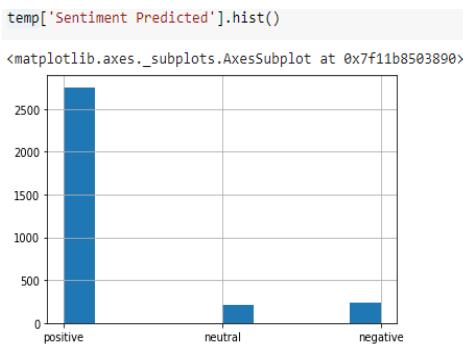


Figure 14: Sentiment Predicted for Amazon Products



Figure 15: Word Cloud Image

5. CONCLUSION

While we assume that a company of yours which has launched recently a product and wishes to learn about market reaction – it is at that juncture the current model will assist you in predicting sentiments by leveraging product reviews and determining how many people have given positive reviews and how many have given negative reviews on one hand. On the other hand, in general the technique that evolves NLP and performed with sentiment analysis it paves a way for NLP researchers to progress in their field of study and witness huge impact. The current study enables one to understand the goal of performing the sentiment analysis using NLP Technique that categorizes customer reviews based on customer sentiments for a product and benefits businesses by tracking customer reviews on a product. The downloaded datasets from Datasets are from Kaggle was used to perform sentiment analysis in order to better understand customer behaviour based on reviews submitted by them. The Python NLTK Library was used to perform various pre-processing techniques, and the Regression Algorithm was used for classification. The model performs well, with F1-Score .86 as evidenced by the model's accuracy matrix. Off late, many companies are using opinion mining to gauge the emotions of the customers connected through their reviews on social media, hence the current study also enables the market to forecast the trend and also strategize their positioning of their products in an effective manner. When the strategies are streamlined as per the customer expectation the companies shall achieve a level of excellence in their quality. The reviews recorded as textual information about a service is a major indicator for service quality and ultimately leading to service excellence. Hence, the current study includes the reviews as dataset of products wherein the services is not taken into consideration. Therefore, the future researchers can take service points into consideration to gauge the textual information.

The study enables to enrich the quality of the products offered by the company incase the corporates initiate suitable customer focused strategies rather than mere product focused strategies. As a result the overall product or service market reaps the fruits of customer experience excellence. The emerging area of CX investment areas are highlighted in the work done by Madan M. Batra., (2017) (a) – they are Overall CX, contact center investment customer insight investment, digital, social and marketing investment respectively. Therefore, the current study also paves a way to explore the possibilities of such areas in corporate or service arenas. Moreover, the study can be helpful at positioning the product or service excellence at an International level too. In future the customer relationship management

system can also be taken into consideration at an advanced level and pitch in a suitable model through the sentiment analysis and opinion mining.

REFERENCES

- [1] Bhatnagar, Anisha. & Bhatia, Madhulika., (2022), 'A Sentiment Analysis Based Approach for Customer Segmentation', Recent Patents on Engineering, Vol:16(2), Pp.32-42(11)
- [2] Enshuo Hsu, Ioannis Malagaris, Yong-Fang Kuo, Rizwana Sultana, Kirk Roberts., (2022), 'Deep learning-based NLP data pipeline for EHR-scanned document information extraction', JAMIA Open, Vol:5(2), Pp:1-12
- [3] Erik Cambria.; Björn Schuller.; Yunqing Xia & Catherine Havasi., (2013)'New Avenues in Opinion Mining and Sentiment Analysis', IEEE Intelligent Systems ,Vol:28(2), Pp:15-21
- [4] Indrawati; Andry Alamsyah.,(2017),'Social network data analytics for market segmentation in Indonesian telecommunications industry', International Conference on Information and Communication Technology (ICoIC7) paper, IAN: 17261950
- [5] Kalpak K.Kulkarni., Arti D.Kalro., Dinesh Sharma & Piyush Sharma., (2020), 'A typology of viral ad sharers using sentiment analysis', Journal of Retailing and Consumer Services , Vol:53,101739
- [6] Kevin Crowston ,Eileen E. Allen &Robert Heckman., (2011) , 'Using natural language processing technology for qualitative data analysis', International Journal of Social Research Methodology, Vol:15(6), Pp: 523-543
- [7] Madan M. Batra., (2017), 'Customer Experience--An Emerging Frontier in Customer Service Excellence', Competition Forum, Pp:198-207
- [8] Madan M. Batra., (2017)a, 'Customer Experience--An Emerging Frontier in Customer Service Excellence', Competition Forum, Pp:198-207
- [9] Md. Rakibul Hasan; Maisha Maliha; M. Arifuzzaman (2019), 'Sentiment Analysis with NLP on Twitter Data', International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)
- [10] Nurfadhlina Mohd Sharef, Harnani Mat Zin and Samaneh Nadali(2016), 'Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data', Journal of Computer Sciences, Vol:12 (3):Pp:153-168
- [11] Simkin, L., (2013), 'To boardrooms and sustainability: the changing nature of segmentation', CentAUR report, Pp:1-48
- [12] Sükrü Ozan All Authors (2018), 'A Case Study on Customer Segmentation by using Machine Learning Methods', International Conference on Artificial Intelligence and Data Processing (IDAP), IAN: 18411438
- [13] Virmani.C., Pillai.A., Juneja.D., (2017), 'Extracting Information from Social Network using NLP', International Journal of Computational Intelligence Research, Vol:13(4), pp. 621-630
- [14] Zhenning Xu., Colin Vail., Amarpreet S. Kohli & Saeed Tajdini., (2021), 'Understanding changes in a brand's core positioning and customer engagement: a sentiment analysis of a brand-owned Facebook site', Journal of Marketing Analytics , Vol: 9, Pp:3-16

Univariate Individual Household Energy Forecasting by Tuned Long Short-Term Memory Network

Marko Stankovic¹[0000–0002–9241–3659], Luka Jovanovic¹[0000–0001–9402–7391],
Milos Antonijevic¹[0000–0002–5511–2531], Aleksandra
Bozovic²[0000–0001–9292–6318], Nebojsa Bacanin¹[0000–0002–2062–924X], and
Miodrag Zivkovic¹[0000–0002–4351–068X]

¹ Singidunum University, Danijelova 32, 11010 Belgrade, Serbia;
marko.stankovic.201@singimail.rs, luka.jovanovic.191@singimail.rs,
mantonijevic@singidunum.ac.rs, nbacanin@singidunum.ac.rs,
mzivkovic@singidunum.ac.rs

² Academy of Applied Technical Studies, 11000 Belgrade, Serbia
abozovic@atssb.edu.rs

Abstract. Accurate forecast of energy consumption has proven to be invaluable to many nodes of the energy sector, enabling an efficient and cost-effective distribution of energy among consumers. However, the nonlinear and non-stationary nature of household energy usage challenges contemporary machine learning algorithms, where there is potential to yet develop robust and dependable technologies. In this paper, we propose a novel, hybrid long short-term memory (LSTM) model tuned by Best Guided Search-Arithmetic Optimization Algorithm (BGS-AOA), that employs quasi-reflection-based learning (QRL) to overcome the exploration-exploitation imbalance of the original AOA. The proposed method was tested on the publicly available dataset capturing energy consumption of individual London households. Due to the stochastic nature of optimizers, all experiments were carried out over the course of five independent runs. In order to evaluate metaheuristics solutions, the total MSE was utilized as the objective function for 3-step ahead forecast. Results of the experiments demonstrated superior performance of our model when compared to other metaheuristics frequently encountered in literature (ABC, FA, SSA, ChOA).

Keywords: Univariate Forecasting · Long Short-Term Memory · Machine Learning · Energy Consumption Forecasting · Metaheuristics

1 Introduction

Energy has been fundamental to the advancement of society given that it is a crucial aspect of human activity. The energy problem, however, has gained significant attention from academics both domestically and internationally due to economy's continued and substantial growth. Predicting energy consumption is

arduous since it depends heavily on a number of variables, including economic growth, national energy legislation, environmental circumstances, and the ability to produce energy [18]. In addition, given the considerable fluctuation and ambiguity involved, predicting individual household electric loads is a rather problematic venture.

The administration, servicing, and functions of the electrical network may all benefit greatly from an efficient projection of the electrical load. Since electricity cannot be deposited in large quantities, there needs to be a reasonable equilibrium between consumption and production [19]. The energy produced and the electricity load must be properly predicted in order to schedule energy activities in an efficient and cost-effective manner. According to the findings of prior research papers, deep learning and machine learning systems are not particularly robust for making predictions [40]. The full scope of these methods for energy use prediction tasks has not been investigated enough and there remains potential for development of dependable technologies.

Therefore, to fill in existing research gap in the domain of energy consumption forecasting, this paper introduces hybrid long short-term memory (LSTM) tuned by metaheuristics for univariate individual households energy consumption forecasting. The LSTM is a robust model for time-series predictions [26, 23], however it should be tuned (optimizer) for each particular challenge in order to obtain satisfying performance. Since there are many LSTM hyper-parameters that may be subjected to optimization process, this tasks is non-deterministic polynomial hard (NP-hard) by nature. It is important to note that no single approach works best for all problems, as outlined by the no free lunch (NFL) theorem of optimization [39], hence experimentation is needed to determine more suitable methods and therefore more powerful models.

The LSTM tuning was performing by using metaheuristics methods, that exhibit outstanding performance for addressing NP-hard problems. For example metaheuristic algorithms simulating mating rituals of firefly's have been capable of addressing complex tasks [8]. Algorithms simulating the swarming behaviors of dragonflies have similarly attained promising performance [50].

For the purpose of this research, an improved version of recently emerged arithmetic optimization algorithm (AOA) is proposed and compared with other cutting-edge methods for LSTM tuning. The AOA has been selected for optimization due the admirable performance shown by the base approach when applied to optimization tasks. However, practical testing suggest that further improvements are possible. Since the LSTM requires substantial amount of computational resources, supplementary goal of this study is to develop a light LSTM structure that consists of maximum 2 LSTM layers, yet achieving solid performance.

The LSTM was evolved by the proposed improved AOA approach and validated against electricity consumption measurements dataset for a sample of 5,567 London households which participated in the UK Power Networks-led Low Carbon London initiative between November 2011 and February 2014.

The remaining of this study is structured in the following way: Section 2 introduces some basic background information related to employed methods along with the literature review, Section 3 provides overview of the basic AOA as well as devised improved method, dataset description used in experiments, simulation setup, results and discussion are given in Section 4, while final remarks of conducted study are given in Section 5.

2 Background and Related Works

The relevant academic literature on forecasting models and methods is included in this chapter. Concerns about the dependability, consistency, efficiency, and precision of the forecasting models are the key obstacles to the subject of forecasting energy use. Forecasting methodologies can be divided into two general classes: (i) model-based, statistical approaches, and (ii) data-driven methods based on machine learning (ML) algorithms [14]. Improving prediction accuracy while reducing the cost function is the common objective of the two.

Classical methods often employ defined standard models and techniques including linear regression, moving average, simple exponential smoothing, and auto-regressive integrated moving average (ARIMA), vector ARIMA (VARIMA), Gaussian processes (GP), to mention a few. They achieve a high degree of accuracy while processing a univariate dataset with bounded, quantifiable, and explainable predictors with relatively minimal computer resources. However, it is probable that conventional time-series approaches cannot detect the nonlinear characteristics of the data and therefore fail to produce accurate predictions. To overcome this limitation, traditional methods may be hybridized with machine learning approaches. Such a model has been proposed by Yuan et al. [42], where the forecast of wind power is achieved with excellent precision by combining the forecast outputs of the least square support vector machine and the auto-regressive fractionally integrated moving average.

Particularly, the characteristics of residential energy usage are nonlinear and non-stationary [38]. Therefore, the focus of this research is on developing an improved, ML-based strategy for forecasting household energy consumption in context of these two factors. Nevertheless, two perplexities persist to be the fundamental barriers to the widespread application of the current data-driven forecasting methodologies [40]. The first factor is that the energy consumption habits of individual households might be highly inconsistent, effectively producing a negative impact on the precision of non-linear data predictions. This volatility is influenced by human behavior [30]. Secondly, as deep learning systems demand multi-dimensional inputs to attain high forecasting accuracy, prediction of univariate time series data, such as energy consumption, is a daunting challenge for these systems. Moreover, the training methods for data-driven models come at a considerable computing time expense. Due to the non-linear properties of energy consumption datasets, developing precise energy prediction technologies remains a challenge for researchers. Hence, ML models that can discover long-

term dependencies in the data and forecast aberrant, rapid changes are necessary [1].

Arguably, the most prominent deep learning method is the Long Short-Term Memory (LSTM) network due to its ability to discern patterns in a long stream of sequential data. According to a review of the literature, hybrid LSTM models typically generate predictions with a superior degree of precision than the singular paradigm. Marino et al. [29] applied LSTM sequence-to-sequence architecture to forecast the electricity consumption for future time steps on the basis of an arbitrary number of previously obtainable load measurements. Single LSTM's predicting performance when processing one-minute resolution dataset was insufficient. Alhussein et al. [4] developed a hybrid CNN-LSTM model to forecast the energy usage of a single household with both normal and volatile consumption patterns. Moradzadeh et al. [32] performed a batch training of a Variational Autoencoder Bidirectional Long Short-Term Memory (VAEBiLSTM) to attain high R values and minimal error calculations when compared to the basic LSTM and support vector regression approach. The CNN-LSTM model suggested by Kim and Cho [28] offers consistent and fast prediction of irregular patterns in household electrical energy usage that could not be anticipated in existing machine learning approaches.

2.1 LSTM Overview

LSTM is an evolved model of the recurrent neural network (RNN) that uses a complex internal structure to process interrelated time-series inputs by learning long-term dependencies[17]. Unlike conventional RNNs, LSTM is specifically designed not to succumb to the vanishing gradient problem[16], where initial signals attenuate as they advance through the propagation chain, preventing the neural network from learning successfully from past information. LSTM has proven to be superior to other contemporary strategies [37][15]. By using a complex four-layer gating structure within its cells, LSTM can successfully achieve long-term memorization via dynamic and individual selection, refinement, and removal of cell's inputs.

The cell state serves the purpose of a memory unit – preserving and transferring meaningful information down the propagation chain. The forget gate controls what to discard from the cell state in order to declutter it from irrelevant information, according to Eq. 1.

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (1)$$

where f_t denotes the forget gate that takes a new input of x_t and old hidden state h_{t-1} , and outputs a value within $(0, 1)$ range, having been compressed by the sigmoid function σ . The corresponding weight matrices are denoted by W_f and U_f , while b_f represents the bias vector. The time step is indicated by the index t .

Selecting new data that will be kept in the cell state is the following step. The input gate acts as a sigmoid layer that decides upon which value to update in the cell state, according to Eq.2.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

where i_t signifies the input gate with a value in the interval $(0, 1)$, W_i and U_i are weight matrices, while b_i denotes the bias vector.

The \tanh layer generates a vector of new candidate values, \tilde{C}_t , as shown in Eq. 3.

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

where W_c , U_c and b_c denote learnable parameters.

In order to update the cell state with actual values, data to be forgotten is first discarded by multiplying the forget gate f_t with the old cell state C_{t-1} , then the new candidate values \tilde{C}_t are scaled by i_t , and added to the former, using Eq. 4.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (4)$$

with C_t being the updated cell state, and \odot signifying the entry-wise product.

The filtered version of the output is generated based on the cell state, as demonstrated in Eq. 5, then further processed by a \tanh layer to produce the new hidden state (Eq. 6).

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (5) \qquad h_t = o_t \odot \tanh(C_t) \quad (6)$$

where o_t denotes the output of the sigmoid layer within $(0, 1)$ interval, with W_o , U_o and b_o being learnable parameters.

2.2 Metaheuristics Optimization

Conventional artificial neural networks (ANNs) use a gradient-based method for training, and in effect suffer from slow convergence, vanishing gradient, and getting trapped in the local minimum, resulting in sub-optimal generalization. Furthermore, the stagnant convergence rate of ANN-based forecasting systems makes them susceptible to over-fitting. Designing a desirable network topology, known as hyperparameter optimization, is a challenging problem due to the requirement that nodes be manually configured by the researcher. A further concern is feature selection, which involves finding the relevant subgroup from high-dimensional attribute sets while eliminating the irrelevant features. Academics refer to this collection of challenges as non-deterministic polynomial (NP)-hard problems. Swarm intelligence models commonly facilitate solutions to these obstacles.

Metaheuristics have seen a booming proliferation in contemporary studies. They were shown to successfully deal with problems of NP-hard complexity by offering a gradient-free approach and avoidance of premature convergence [43].

Oliva et al. [33] applied artificial bee colony (ABC) algorithm to precisely determine the parameters of solar cells' performance. The method demonstrated reliability and precision in solving complex non-linear and multi-modal objective functions. Other very successful use cases range from COVID-19 cases forecasting and related applications [46, 49, 12], cloud computing optimization [48, 11], wireless sensor network optimization [45, 47, 50], feature selection problem [10], image classification in medical and other domains [44, 35, 13, 24], credit card fraud detecting task [20, 34], pollution forecasting [6], general network security [9, 36, 21], and the general tuning of a variety of machine learning models including LSTM [7, 23, 22, 3].

3 Proposed Method

3.1 Original Arithmetic Optimization Algorithm

The AOA makes use of a population of agents and stochastic mechanisms to tackle search problems. The two principal phases of the optimization process—exploration and exploitation—are shared by all metaheuristics. The exploration phase is responsible for searching through the unfamiliar regions of the search space, whereas the exploitation phase is responsible for focusing on the regions that have already been investigated. The AOA is a contemporary population-based metaheuristic method introduced by Abualigah et al.[2] that excludes the need for derivation when solving optimization tasks.

Basic search procedure of AOA is modeled with following equations:

$$MOA(C_Iter) = \text{Min} + C_Iter \times \left(\frac{\text{Max} - \text{Min}}{M_Iter} \right) \quad (7)$$

$$x_{i,j}(C_Iter + 1) = \begin{cases} \text{best } (x_j) \div (MOP + \epsilon) \times ((UB_j - LB_j) \times \mu + LB_j), & r2 < 0.5 \\ \text{best } (x_j) \times MOP \times ((UB_j - LB_j) \times \mu + LB_j), & \text{otherwise} \end{cases} \quad (8)$$

$$MOP(C_Iter) = 1 - \frac{C_Iter^{1/\alpha}}{M_Iter^{1/\alpha}} \quad (9)$$

$$x_{i,j}(C_Iter + 1) = \begin{cases} \text{best } (x_j) - MOP \times ((UB_j - LB_j) \times \mu + LB_j), & r3 < 0.5 \\ \text{best } (x_j) + MOP \times ((UB_j - LB_j) \times \mu + LB_j), & \text{otherwise} \end{cases}, \quad (10)$$

where $MOA(C_Iter)$ is function value in the current iteration, spanning between Min and Max , the M_Iter and $[1, (M_Iter)]$ denote maximum iterations in the run and current iteration in the range, respectively, $r1, r2, r3$ denote pseudo-random numbers drawn from the unifrom distribution within the range $[-1, 1]$, ϵ is small integer, x denotes solution, while i and j represent individual number and parameter number, respectively. Finally, LB_j, UB_j denote j th position's lower and upper limit, respectively, while MOP and α stand for the math

optimizer probability and constant parameter evaluation exploitation accuracy, respectively.

More details about basic AOA can be captured from [2].

3.2 Enhanced AOA metaheuristics

Notwithstanding satisfying AOA performance for real-world NP-hard challenges [2], during practical observations basic AOA inner workings, some issues that need to be addressed are noticed. It happens in some runs that the final outputs are not satisfying because the search process tends to be biased more towards exploitation or exploration. It seems that *MOA* and *MOP* operators are not robust enough to guarantee exploitation-exploration balance in all runs.

Therefore, to overcome above mentioned deficiencies, proposed method introduces best guided search (BSG) mechanism, that executes in two models, in the original AOA. In early iterations, with the assumption that search space region where an optimum (sub-optimum) solution is located has not yet been discovered, the worst solution is replaced with the quasi-reflective opposite individual $best_{x_qrl}$ of the current best solution, which is generated by applying quasi-reflection-based learning (QRL) procedure. It was shown in previous studies that the QRL effectively generates solution in the opposite part of the search space [5].

However, in later iterations, when the optimum region of the search space is found, the worst agent is replaced with the solution x_u , that is derived by applying uniform crossover operator between two best individual from the population. These two behavior are controlled with hard-coded control parameter best guided search mode (*bsgm*) which is set to $M_{iter}/2$. This value was determined empirically.

Finally, devised method is named BSG-AOA and its pseudo-code and process flowchart are shown below.

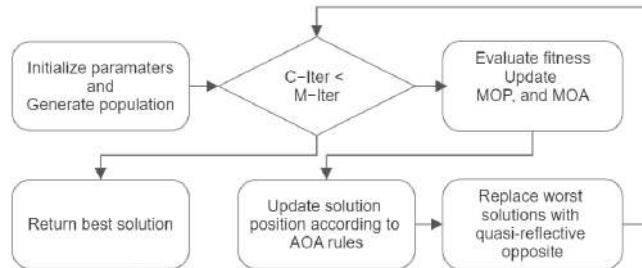


Fig. 1: Enhanced AOA algorithm flowchart

Algorithm 1 The BSG-AOA pseudo-code

```

Initialization of the parameters  $\alpha$  and  $\mu$ .
Produce the individuals in the population arbitrarily ( $i = 1, \dots, N$ ).
while  $C\_Iter < M\_Iter$  do
    Compute the fitness function values for each solution.
    Determine the best solution until now.
    Revise MOA value by utilizing Eq. 7.
    Revise MOP value by utilizing Eq. 9.
    for  $i = 1$  to  $Solutions$  do
        for  $j = 1$  to  $Positions$  do
            Produce an arbitrary number ( $r1, r2, r3$ ) in interval [0,1].
            if  $r1 > MOA$  then
                Exploration phase
                if  $r2 > 0.5$  then
                    Employ the Division operator ( $D$ , " $\div$ ")
                    Revise the position of the  $i$ th individual by utilizing the first rule in Eq. 8.
                else
                    Employ the Multiplication operator ( $M$ , " $\times$ ")
                    Revise the position of the  $i$ th individual by utilizing the second rule in Eq. 8.
                end if
            else
                Exploitation phase
                if  $r3 > 0.5$  then
                    Employ the Subtraction operator ( $S$ , " $-$ ")
                    Revise the position of the  $i$ th individual by utilizing the first rule in Eq. 10.
                else
                    Employ the Addition operator ( $A$ , " $+$ ")
                    Revise the position of the  $i$ th individual by utilizing the second rule in Eq. 10.
                end if
            end if
        end for
    end for
    if  $beg < M_{iter}$  then
        Replace  $worst_x$  with  $best_{x_{qri}}$ 
    else
        Replace  $worst_x$  with  $x_u$ 
    end if
     $C\_Iter = C\_Iter + 1$ 
end while
Return the best individual.

```

4 Experiments and Discussion

This section first provides an overview of the dataset used in experiments followed by basic experimental setup, comparative analysis and discussion.

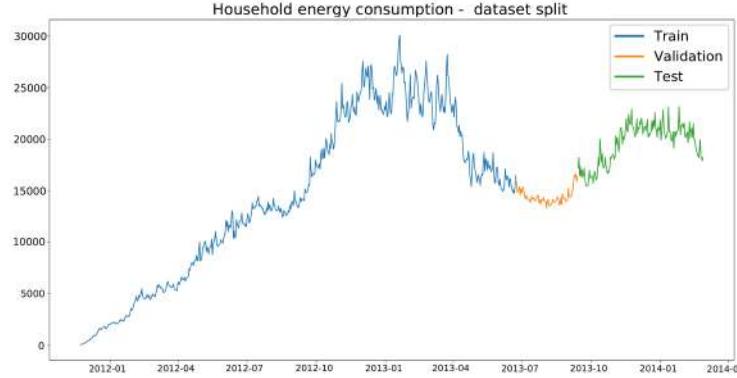
4.1 Dataset

The dataset utilized in this research originates from the London Data Store, and encompasses the electricity consumption measurements for a sample of 5,567 London households which participated in the UK Power Networks-led Low Carbon London initiative between November 2011 and February 2014. The data has been captured using smart power meters, and has a daily resolution with a total of 19752 samples. This dataset is publicly available on Kaggle: <https://www.kaggle.com/code/rheajgurung/energy-consumption-forecast/notebook>. Individual households' measurements of electricity usage over the course of 829 days were aggregated, and employed as the target variable for univariate time-series predictions. The dataset is divided into training, validation and testing according to 70%, 10%, 20% split, as shown in Figure 2.

4.2 Experimental setup, results and discussion

All tested models were independently implemented in Python, using TensorFlow and Keras. The prediction model used six input steps, representing six days worth of data, used to predict energy consumption three 3 (days) ahead.

Fig. 2: Electricity consumption dataset split



The proposed BGS-AOA was compared against the ABC [25], FA [41], SSA [31], and chimp swarm optimization algorithm (ChOA) [27]. In the results summary, all algorithms adopted for LSTM tuning have prefix 'LSTM'. Throughout testing, all algorithms employed a population of five individuals, which were improved over 5 iterations. All experiments were conducted in 5 independent runs due to the stochastic nature of optimizers.

Every metaheuristics individual represents an array of 6 LSTM hyper-parameters which were optimized. Parameters taken into account along with its boundaries are as follows: number of neurons in the first layer ($nn1$) [100, 200], learning rate (lr) [0.0001, 0.01], number of training epochs ($epochs$) [300, 600], dropout (dp) [0.001, 0.01], number of layers (ln) [1, 2] and number of neurons in the second layer ($nn2$) [100, 200]. Former parameter was only used if the generated ln value is 2. All models utilized Adam optimizer and mean square error (MSE) loss function.

Throughout experiments the following metrics were captured for each prediction step separately, but also calculated as the overall results for each 3 steps: coefficient of determination R^2 , MSE, mean absolute error (MAE), MSE and root MSE (RMSE). The overall MSE was used as objective function for metaheuristics solutions' evaluation. It is noted that since the LSTM is neural network, all values in the dataset were scaled within the range [0, 1].

The overall objective function (MSE) over 5 independent runs in terms of best, worst, mean, median, standard deviation and variance is shown in Table 1. Tables 2 and 3 exhibit normalized and denormalized metrics for the best run, respectively and the best obtained LSTM hyper-parameters are shown in Table 4.

From the presented tables it can be clearly concluded that the proposed LSTM-BSG-AOA exhibits the best performance, for overall as for the individual metrics. It achieved the best scores for the best and median results, followed by the LSTM-ChOA on the second place, as seen in Table 1. It also achieved superior scores with respect to the detailed metrics of the best produced LSTM, as given

Table 1: Overall objective function metrics over 5 runs for 3-steps ahead forecast

Method	Best	Worst	Mean	Median	Std	Var
LSTM-BGS-AOA	9.942E-04	1.011E-03	9.984E-04	9.957E-04	6.248E-06	3.904E-11
LSTM-ABC	1.001E-03	1.004E-03	1.003E-03	1.004E-03	1.567E-06	2.456E-12
LSTM-FA	1.002E-03	1.021E-03	1.008E-03	1.005E-03	6.811E-06	4.639E-11
LSTM-SSA	9.966E-04	1.003E-03	9.988E-04	9.978E-04	2.471E-06	6.105E-12
LSTM-ChOA	9.970E-04	9.979E-04	9.973E-04	9.973E-04	3.183E-07	1.013E-13

Table 2: The R2, MAE, MSE, and RMSE normalized metrics of best-generated LSTM with 3 steps ahead forecast

	Performance indicator	LSTM-BGS-AOA	LSTM-ABC	LSTM-FA	LSTM-SSA	LSTM-ChOA
One-step ahead	R2	0.740888	0.739594	0.739939	0.740399	0.741392
	MAE	0.024895	0.025172	0.025266	0.024931	0.025047
	MSE	0.001002	0.001007	0.001006	0.001004	0.001000
	RMSE	0.031660	0.031739	0.031718	0.031690	0.031629
Two-step ahead	R2	0.743426	0.740904	0.740356	0.742037	0.743008
	MAE	0.024784	0.025199	0.025286	0.024876	0.024969
	MSE	0.000993	0.001002	0.001004	0.000998	0.000994
	RMSE	0.031505	0.031659	0.031693	0.031590	0.031530
Three-step ahead	R2	0.744724	0.743581	0.742627	0.744684	0.742455
	MAE	0.024781	0.025006	0.025144	0.024728	0.025131
	MSE	0.000988	0.000992	0.000996	0.000988	0.000996
	RMSE	0.031425	0.031495	0.031554	0.031427	0.031564
Overall Results	R2	0.743013	0.741360	0.740974	0.742373	0.742285
	MAE	0.024820	0.025126	0.025232	0.024845	0.025049
	MSE	0.000994	0.001001	0.001002	0.000997	0.000997
	RMSE	0.031530	0.031631	0.031655	0.031569	0.031575

Table 3: The R2, MAE, MSE, and RMSE denormalized metrics of best-generated LSTM with 3 steps ahead forecast

	Error indicator	LSTM-BGS-AOA	LSTM-ABC	LSTM-FA	LSTM-SSA	LSTM-ChOA
One-step ahead	R2	0.741	0.739	0.739	0.740	0.741
	MAE	747.939	756.259	759.078	749.023	752.511
	MSE	904768.060	909284.129	908080.324	906475.095	903006.692
	RMSE	951.193	953.564	952.932	952.089	950.267
Two-step ahead	R2	0.743	0.741	0.740	0.742	0.743008
	MAE	744.593	757.079	759.687507	747.358	750.149
	MSE	895906.215	904710.219	906626.395	900756.874	897364.470
	RMSE	946.523	951.162	952.169	949.082	947.293
Three-step ahead	R2	0.745	0.743	0.743	0.744	0.742455
	MAE	744.529	751.276	755.427	742.934	755.027
	MSE	891371.669	895365.385	898695.052	891511.448	899295.819
	RMSE	944.125	946.237	947.995	944.199	948.312
Overall Results	R2	0.743	0.741	0.741	0.742	0.742285
	MAE	745.687	754.871	758.064	746.438	752.562
	MSE	897348.648	903119.911	904467.257	899581.139	899888.994
	RMSE	947.285	950.326	951.035	948.462	948.625

in Tables 2 and 3. Finally, the best set of hyperparameters determined by the LSTM-BSG-AOA approach was: 155 neurons in the first layer, learning rate of

Table 4: Parameters of best solutions for each metaheuristics

Method	<i>nn1 lr</i>	<i>epoch</i> s	<i>dp</i>	<i>ln nn2</i>
LSTM-BGS-AOA	157	0.005108	533	0.095663 2 103
LSTM-ABC	159	0.007818	491	0.118743 1 190
LSTM-FA	100	0.010000	500	0.050000 1 176
LSTM-SSA	102	0.008604	591	0.077705 2 174
LSTM-ChOA	156	0.010000	502	0.150209 1 119

0.005108, 533 epochs, dropout value of 0.095663, 2 layers total and 103 neurons in the second layer.

To visualize results of all metaheuristics, kernel density estimation (KDE) plots for objective function over 5 runs and the best LSTM-BSG-AOA results are depicted in Figure 3. Following the KDE plots is the graph showing recorded one, two and three step predictions made by the best performing networks compared to actual values in the dataset once given 6 input variables from a never before seen set of testing data.

5 Conclusion

The research presented in this paper covers a novel LSTM-based method for time-series forecasting of individual household energy consumption. To more efficiently tackle this task a novel improved version of the AOA metaheuristic algorithms is proposed so as to overcome certain known shortcomings of the original. The proposed algorithms has been dubbed the BSG-AOA, and applied to optimizing LSTM networks structures to better address the given problem. The novel proposed metaheuristic, has been tasked with optimizing hyperparameters withing a LSTM netowrk, and appropriately named LSTM-BSG-AOA. The performance of which has been compared against four other state-of-the-art algorithms including: ABC, FA, SSA and ChOA, applied to the same task. While high computational demands present a limitation when testing all LSTM models the conducted simulation outcomes for one-step, two-step and three-step ahead forecast, as well as the overall results evidently show the superiority of the suggested LSTM-BSG-AOA for the given forecasting task.

Future work will focus on further exploring the potential of the proposed approach on other energy related time-series prediction problems, as well as exploring the potential in real time applications. Additionally, authors plan on exploring the potential of other ML models optimized by emerging metaheuristics in hopes of further improving performance on the energy consumption forecasting tasks.

References

1. Highly accurate energy consumption forecasting model based on parallel lstm neural networks. *Advanced Engineering Informatics* **51**, 101442– (2022).

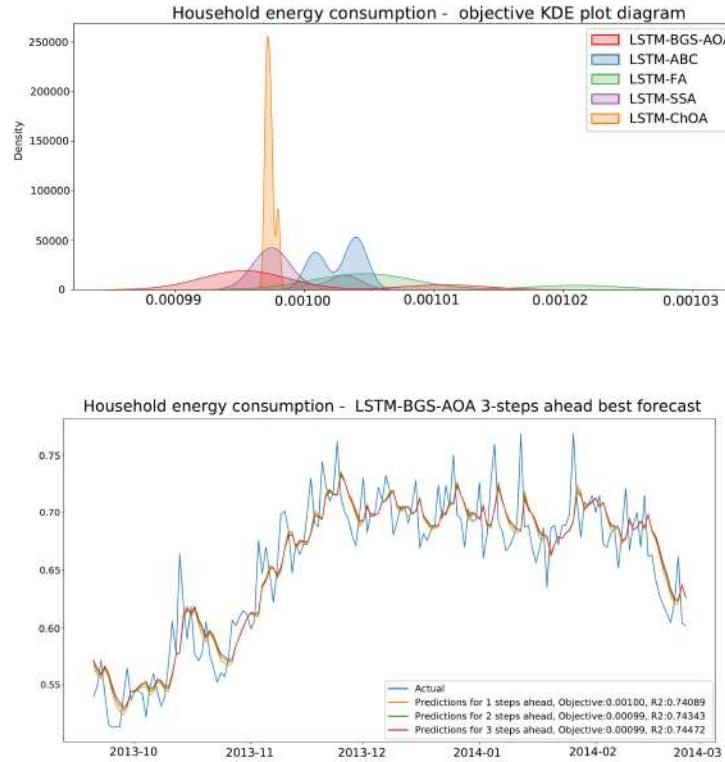


Fig. 3: KDE plot for objective function over 5 runs (top), best predicted results by LSTM-BSG-AOA (bottom)

- <https://doi.org/10.1016/J.AEI.2021.101442>
2. Abualigah, L., Diabat, A., Mirjalili, S., Elsayed Abd Elaziz, M., Gandomi, A.: The arithmetic optimization algorithm. Computer Methods in Applied Mechanics and Engineering **376**, 113609 (04 2021). <https://doi.org/10.1016/j.cma.2020.113609>
 3. AlHosni, N., Jovanovic, L., Antonijevic, M., Bukumira, M., Zivkovic, M., Strumberger, I., Mani, J.P., Bacanin, N.: The xgboost model for network intrusion detection boosted by enhanced sine cosine algorithm. In: International Conference on Image Processing and Capsule Networks. pp. 213–228. Springer (2022)
 4. Alhussein, M., Khursheed, K., Haider, S.: Hybrid cnn-lstm model for short-term individual household load forecasting. IEEE Access **8**, 180544–180557 (01 2020). <https://doi.org/10.1109/ACCESS.2020.3028281>
 5. Bacanin, N., Bezdan, T., Venkatachalam, K., Zivkovic, M., Strumberger, I., Abouhawwash, M., Ahmed, A.B.: Artificial neural networks hidden unit and weight connection optimization by quasi-refection-based learning artificial bee colony algorithm. IEEE Access **9**, 169135–169155 (2021)

6. Bacanin, N., Sarac, M., Budimirovic, N., Zivkovic, M., AlZubi, A.A., Bashir, A.K.: Smart wireless health care system using graph lstm pollution prediction and dragonfly node localization. *Sustainable Computing: Informatics and Systems* **35**, 100711 (2022)
7. Bacanin, N., Stoean, C., Zivkovic, M., Jovanovic, D., Antonijevic, M., Mladenovic, D.: Multi-swarm algorithm for extreme learning machine optimization. *Sensors* **22**(11), 4204 (2022)
8. Bacanin, N., Stoean, R., Zivkovic, M., Petrovic, A., Rashid, T.A., Bezdan, T.: Performance of a novel chaotic firefly algorithm with enhanced exploration for tackling global optimization problems: Application for dropout regularization. *Mathematics* **9**(21), 2705 (2021)
9. Bacanin, N., Zivkovic, M., Stoean, C., Antonijevic, M., Janicijevic, S., Sarac, M., Strumberger, I.: Application of natural language processing and machine learning boosted with swarm intelligence for spam email filtering. *Mathematics* **10**(22), 4173 (2022)
10. Bezdan, T., Cvetnic, D., Gajic, L., Zivkovic, M., Strumberger, I., Bacanin, N.: Feature selection by firefly algorithm with improved initialization strategy. In: *7th Conference on the Engineering of Computer Based Systems*. pp. 1–8 (2021)
11. Bezdan, T., Zivkovic, M., Tuba, E., Strumberger, I., Bacanin, N., Tuba, M.: Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm. In: *International Conference on Intelligent and Fuzzy Systems*. pp. 718–725. Springer (2020)
12. Budimirovic, N., Prabhu, E., Antonijevic, M., Zivkovic, M., Bacanin, N., Strumberger, I., Venkatachalam, K.: Covid-19 severity prediction using enhanced whale with salp swarm feature classification. *Computers, Materials and Continua* pp. 1685–1698 (2022)
13. Bukumira, M., Antonijevic, M., Jovanovic, D., Zivkovic, M., Mladenovic, D., Kunjadic, G.: Carrot grading system using computer vision feature parameters and a cascaded graph convolutional neural network. *Journal of Electronic Imaging* **31**(6), 061815 (2022)
14. Cao, X., Dong, S., Wu, Z., Jing, Y.: A data-driven hybrid optimization model for short-term residential load forecasting. pp. 283–287 (10 2015). <https://doi.org/10.1109/CIT/IUCC/DASC/PICOM.2015.41>
15. He, W.: Load forecasting via deep neural networks. *Procedia Computer Science* **122**, 308–314 (01 2017). <https://doi.org/10.1016/j.procs.2017.11.374>
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (Nov 1997)
17. Hochreiter, S.: Studies on dynamic neural networks. Master's thesis, Institute for Computer Science, Technical University, Munich **1**, 1–150 (1991)
18. Hu, H., Wang, L., Lv, S.X.: Forecasting energy consumption and wind power generation using deep echo state network. *Renewable Energy* **154**, 598–613 (2020)
19. Jetcheva, J.G., Majidpour, M., Chen, W.P.: Neural network model ensembles for building-level electricity load forecasts. *Energy and Buildings* **84**, 214–223 (2014)
20. Jovanovic, D., Antonijevic, M., Stankovic, M., Zivkovic, M., Tanaskovic, M., Bacanin, N.: Tuning machine learning models using a group search firefly algorithm for credit card fraud detection. *Mathematics* **10**(13), 2272 (2022)
21. Jovanovic, D., Marjanovic, M., Antonijevic, M., Zivkovic, M., Budimirovic, N., Bacanin, N.: Feature selection by improved sand cat swarm optimizer for intrusion detection. In: *2022 International Conference on Artificial Intelligence in Everything (AIE)*. pp. 685–690. IEEE (2022)

22. Jovanovic, L., Antonijevic, M., Zivkovic, M., Jovanovic, D., Marjanovic, M., Bacanin, N.: Sine cosine algorithm with tangent search for neural networks dropout regularization. In: Data Intelligence and Cognitive Informatics, pp. 789–802. Springer (2023)
23. Jovanovic, L., Jovanovic, D., Bacanin, N., Jovanca Stakic, A., Antonijevic, M., Magd, H., Thirumalaisamy, R., Zivkovic, M.: Multi-step crude oil price prediction based on lstm approach tuned by salp swarm algorithm with disputation operator. *Sustainability* **14**(21), 14616 (2022)
24. Jovanovic, L., Zivkovic, M., Antonijevic, M., Jovanovic, D., Ivanovic, M., Jassim, H.S.: An emperor penguin optimizer application for medical diagnostics. In: 2022 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC). pp. 191–196. IEEE (2022)
25. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. *Journal of global optimization* **39**(3), 459–471 (2007)
26. Kavevan, Z., Suykens, J.A.: Transductive lstm for time-series prediction: An application to weather forecasting. *Neural Networks* **125**, 1–9 (2020)
27. Khishe, M., Mosavi, M.R.: Chimp optimization algorithm. *Expert systems with applications* **149**, 113338 (2020)
28. Kim, T.Y., Cho, S.B.: Predicting residential energy consumption using cnn-lstm neural networks. *Energy* **182** (06 2019). <https://doi.org/10.1016/j.energy.2019.05.230>
29. Marino, D., Amarasinghe, K., Manic, M.: Building energy load forecasting using deep neural networks (10 2016). <https://doi.org/10.1109/IECON.2016.7793413>
30. Masson, V., Hidalgo, J., Amossé, A., Bélaïd, F., Bocher, E., Bonhomme, M., Bourgeois, A., Bretagne, G., Caillerez, S., Cordeau, E., Demazeux, C., Faraut, S., Gallato, C., Haouès-Jouve, S., Lambert, M.L., Lemonsu, A., Lestringant, R., Lévy, J.P., Long, N., Vye, D.: Urban climate, human behavior & energy consumption: from lcz mapping to simulation and urban planning (the mapuce project) (07 2015)
31. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in engineering software* **114**, 163–191 (2017)
32. Moradzadeh, A., Moayyed, H., Zare, K., Mohammadi-ivatloo, B.: Short-term electricity demand forecasting via variational autoencoders and batch training-based bidirectional long short-term memory. *Sustainable Energy Technologies and Assessments* **52**, 102209 (04 2022). <https://doi.org/10.1016/j.seta.2022.102209>
33. Oliva, D., Cuevas, E., Pajares, G.: Parameter identification of solar cells using artificial bee colony optimization. *Energy* **72**, 93–102 (2014)
34. Petrovic, A., Bacanin, N., Zivkovic, M., Marjanovic, M., Antonijevic, M., Strumberger, I.: The adaboost approach tuned by firefly metaheuristics for fraud detection. In: 2022 IEEE World Conference on Applied Intelligence and Computing (AIC). pp. 834–839. IEEE (2022)
35. Prakash, S., Kumar, M.V., Ram, S.R., Zivkovic, M., Bacanin, N., Antonijevic, M.: Hybrid glfil enhancement and encoder animal migration classification for breast cancer detection. *Comput. Syst. Eng.* **41**(2), 735–749 (2022)
36. Stankovic, M., Antonijevic, M., Bacanin, N., Zivkovic, M., Tanaskovic, M., Jovanovic, D.: Feature selection by hybrid artificial bee colony algorithm for intrusion detection. In: 2022 International Conference on Edge Computing and Applications (ICECAA). pp. 500–505. IEEE (2022)

37. Wei, N., Li, C., Duan, J., Liu, J., Zeng, F.: Daily natural gas load forecasting based on a hybrid deep learning model. *Energies* **12**, 15 (01 2019). <https://doi.org/10.3390/en12020218>
38. Wei, S., Bai, X.: Multi-step short-term building energy consumption forecasting based on singular spectrum analysis and hybrid neural network. *Energies* **15**, 1743 (02 2022). <https://doi.org/10.3390/en15051743>
39. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* **1**(1), 67–82 (1997)
40. Yan, K., Li, W., Ji, Z., Qi, M., Du, Y.: A hybrid lstm neural network for energy consumption forecasting of individual households. *IEEE Access* **7**, 157633–157642 (2019). <https://doi.org/10.1109/ACCESS.2019.2949065>
41. Yang, X.S., Slowik, A.: Firefly algorithm. In: *Swarm Intelligence Algorithms*, pp. 163–174. CRC Press (2020)
42. Yuan, X., Tan, Q., Yuan, Y., Wu, X.: Wind power prediction using hybrid autoregressive fractionally integrated moving average and least square support vector machine. *Energy* **129** (04 2017). <https://doi.org/10.1016/j.energy.2017.04.094>
43. Zhang, J., Xiao, M., Gao, L., Pan, Q.K.: Queuing search algorithm: A novel meta-heuristic algorithm for solving engineering optimization problems. *Applied Mathematical Modelling* **63** (07 2018). <https://doi.org/10.1016/j.apm.2018.06.036>
44. Zivkovic, M., Bacanin, N., Antonijevic, M., Nikolic, B., Kvascev, G., Marjanovic, M., Savanovic, N.: Hybrid cnn and xgboost model tuned by modified arithmetic optimization algorithm for covid-19 early diagnostics from x-ray images. *Electronics* **11**(22), 3798 (2022)
45. Zivkovic, M., Bacanin, N., Tuba, E., Strumberger, I., Bezdan, T., Tuba, M.: Wireless sensor networks life time optimization based on the improved firefly algorithm. In: *2020 International Wireless Communications and Mobile Computing (IWCMC)*. pp. 1176–1181. IEEE (2020)
46. Zivkovic, M., Bacanin, N., Venkatachalam, K., Nayyar, A., Djordjevic, A., Strumberger, I., Al-Turjman, F.: Covid-19 cases prediction by using hybrid machine learning and beetle antennae search approach. *Sustainable Cities and Society* **66**, 102669 (2021)
47. Zivkovic, M., Bacanin, N., Zivkovic, T., Strumberger, I., Tuba, E., Tuba, M.: Enhanced grey wolf algorithm for energy efficient wireless sensor networks. In: *2020 Zooming Innovation in Consumer Technologies Conference (ZINC)*. pp. 87–92. IEEE (2020)
48. Zivkovic, M., Bezdan, T., Strumberger, I., Bacanin, N., Venkatachalam, K.: Improved harris hawks optimization algorithm for workflow scheduling challenge in cloud–edge environment. In: *Computer Networks, Big Data and IoT*, pp. 87–102. Springer (2021)
49. Zivkovic, M., Venkatachalam, K., Bacanin, N., Djordjevic, A., Antonijevic, M., Strumberger, I., Rashid, T.A.: Hybrid genetic algorithm and machine learning method for covid-19 cases prediction. In: *Proceedings of International Conference on Sustainable Expert Systems: ICSES 2020*. vol. 176, p. 169. Springer Nature (2021)
50. Zivkovic, M., Zivkovic, T., Venkatachalam, K., Bacanin, N.: Enhanced dragonfly algorithm adapted for wireless sensor network lifetime optimization. In: *Data intelligence and cognitive informatics*, pp. 803–817. Springer (2021)

Interpreting Doctor's Handwritten Prescription Using Deep Learning Techniques

Rizwanullah Mohammad, Ajay Kumar Varma Nagaraju and Suneetha Manne

<https://orcid.org/0000-0002-8917-276X>

IT Department, Velagapudi Ramakrishna Siddhartha Engineering
College, Vijayawada, Andhra Pradesh, India

<mailto:mailtorizwanullah@gmail.com>

ajaynagaraju32@gmail.com

hodit@vrsiddhartha.ac.in

* Corresponding Author : Rizwanullah Mohammad

Abstract: A Doctor's Handwriting Recognition model can predict (recognize) the text present in the doctor's prescription, by feeding image of that medicine name to the model and it predicts the text present in the image and it gives the final medicine name as digital text. This model is suitable only for Text written in English Language and not suitable for other languages of texts written in prescription. The model based on training dataset the output it produces may get varied and based on training images count. Both convolution layers and Bi-Directional LSTM layers can be used for feature extraction and recognizing text respectively.

Keywords: Bi-Directional LSTM units, Bi-Directional Gated Recurrent Units, Convolution Layers, Adam optimizer, Relu Activation Function.

1 Introduction

It is most common that people can't understand and interpret the doctor's handwriting. The calligraphy they follow which is always challenging for ordinary people and even for pharmacist to understand doctor's handwriting [1-3]. Until they understand correctly then cant give correct medicine to Patient. Due to usage of wrong medicines they may face severe consequences with respect to their health. This problem need to be solved with the latest technologies those are present[5]. The solution for this is deep learning models[11]. A deep learning model can take large data input and can process with help of neural network and layers[13]. They can give high accuracy and more reliable. Now by making use of deep learning techniques, involving all the terms in deep learning to provide a optimal solution for this. The Bi-Directional LSTM model can provide a solution which can predict text present doctor's prescription's image which are passed as input to our model. The comparision was made between Bi-Directional LSTM and Bi-Directional

GRU[17]. For this application of interpreting doctor's handwritten prescriptions, either one of the architectures between Bi-LSTM and Bi-GRU will be chosen and implemented[23].IAM is a dataset which contains handwritings of many people. The two algorithms, Bi-Directional LSTM and Bi-Directional GRU, are trained using images from the IAM Dataset, and the accuracy results are compared and examined. This paper provides a brief explanation of how the performance of model is affected by the purity and amount size of the dataset.

2 Literature Study

Here are the available models, That Researchers proposed and developed. They Proposed different methodologies to recognize text in an image and showcasing their performance.

T. Jain et al. [1] presented a model employing the BI-LSTM Model for the recognition of a doctor's handwriting. They have only created a model; no mobile or web applications have been created to execute the model in real time. In order to minimise overfitting and increase the model's resistance to noise, they employed data augmentation approaches. Changing network topologies increases accuracy and reduces complexity.

S. Tabassum et al. [2] presented a model for the identification of doctors' handwriting. After employing the SRP Augmentation approach, they achieved an accuracy of 89%. Some of the participating physicians' prescription pictures were made available. They made a dataset called handwritten corpus. The introduction of SRP, increased the size of the data sets. For predicting the handwriting of doctors, an online character recognition system utilising Bi-LSTM was employed. Variable handwriting styles should be added in the dataset to improve the model's knowledge and prediction abilities.

L. J. Fajardo et al. [3] presented a model for interpreting the doctor's handwriting was developed. They employed CRNN Model. The accuracy of the tests conducted using the mobile application was 72%. The model is implemented through the use of a mobile application called DCHRS and aims to recognise the name of the medication inside prescription that has been captured, as well as to provide the digital text of the handwriting. Low image count and precision are insufficient for medical applications.

España-Boquera et al. [4] presented a public dataset named SPA sentences. And provided indicators at level of sentences, proposed methodology of combining convolutional blocks with LSTM and CTC blocks. Unidirectional LSTM is less accurate than Bidirectional LSTM.

Maalej et al. [5] proposed a methodology which replaces 1-D LSTM with Multi LSTM Blocks. They connected MDLSTM blocks with CTC-Maxout block and they performed Data augmentation which increases accuracy. MDLIST increases model complexity and the interpretability will get reduced.

A. Harikrishnan et al.[6] presented their model for hand-written numbers recognition with CNN and MLP Architectures. The model was trained with CNN and MLP

Architectures of models. The results are, CNN achieved 99% Accuracy and MLP achieved 97% Accuracy. CNN performance is more than MLP. According to the proposed model if the number of CNN layers is increased then it may give better Accuracy.

A. Nikitha et al. [7] proposed a handwritten text recognition system which uses word error rate instead of character error rate. The Two dimensional LSTM Architecture is used for model. The model wasn't evaluated on character error rate, a crucial criterion for making it more robust and efficient.

S. Hassan et al. [9] presented a methodology for recognition of handwritten text for any script. Their proposed model achieved an average accuracy of 83%. The proposed architecture's network topology must be adjusted to raise the number of filters to 1024 for better results.

Ul Sehr Zia et al [10] presented a handwritten text recognition model based on CRNN. This model was trained with NUST Urdu handwriting dataset. And demonstrated how model can be trained with English and Urdu bilingual handwriting. Alrobah, N et al. [11] proposed applying of robust methodologies to handle challenges in recognition of different language handwriting data. Mainly the Arabic handwritten data. The methods to use and how to make feature extractions in most effective way which can help model to be more Robust and Perform well.

U. Shaw et al. [12] proposed a model which can recognize poor legible handwriting and provide a digital text. OCR was used in order to recognize handwriting. The model was trained with MNIST dataset. Generally, for handwriting recognition the model should be trained with dataset containing alphabets rather than numbers.

S. Sharma et al. [13] demonstrated the usage of various scripts like Devanagari, Gurmukhi and other language datasets to train the model. A model was developed with CNN architecture. RNNs are recommended for successful model feature decoding.

Sethy et al. [14] demonstrated the CNN architecture usage for efficient feature extraction and pattern recognition. The system was training with Odia and Bangla Handwritten data. This data was trained with CNN Architecture. RNN architecture must be employed to recognize text.

S. Haboubi et al. [17] presented a model for Urdu handwriting recognition. They demonstrated how the Bi-GRUs are giving better accuracy with less complexity and consuming less memory. Bi-GRU is LSTM's rival.

A. Abdallah et al [23] presented hand-writing recognition model for Russian scripts. They implemented CNN for feature extraction and Multi-dimensional GRU for feature decoding. They confirmed Multi-dimensional GRU's good performance

The above methods and presentations gives us an in-depth grasp of their models, performance, and design approaches. By understanding their approaches, better ones can be devised to overcome obstacles and design a better model.

3 Proposed Work

3.1 Dataset

The suggested model used 86800 grayscale pictures. IAM Dataset contains different handwriting scripts[16].The dataset is splitted into 90:5:5 ratio(Train_Validate_Test).

3.2 Pre-processing

In this pre-processing phase the images will get reshaped to width of 128 and height of 32 and padding to 99. And then the datatype will be changed to float32 which is called as casting. This may increase the model performance.This is distortion free image processing. Initial, using tf.image is the first stage in the resizing process. The image path should be sent along with width, and height to the resize() function while maintaining the aspect ratio. Following this, the additional padding is added to resized image. Padding can be added to image by subtracting width and height values(128,32 respectively) with image shape which image want to add padding to, with help of tf.shape() function. With tf.transpose() function by giving perm = [1,0,2] This is nothing but setting up the required tensor dimensions. Perform flip_left_right of image inorder to get image flipped along with width dimensions.

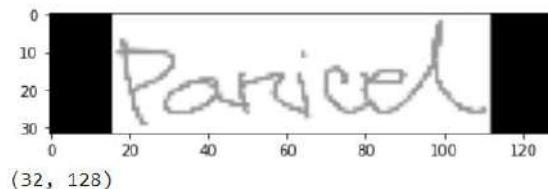


Fig. 1. Distortion free image pre-processing

3.3 Design Methodology

A brief description of proposed methodology:

The cropped images from the IAM Dataset are being utilized and a few medical datasets in conjunction with our model to train. The training dataset, validation dataset, and testing datasets have been divided in a ratio of 90:5:5.The network has a sophisticated design that includes seven convolutional layers, optional batch Normalization layers, Max Pooling layers, Relu activation functions, a Bi-directional LSTM layer, and a CTC layer[1].The procedure increases the number of channels in the first convolution layer from 1 to 64. which, after several layers, is raised to 128.

The below diagram describes architecture of our work.

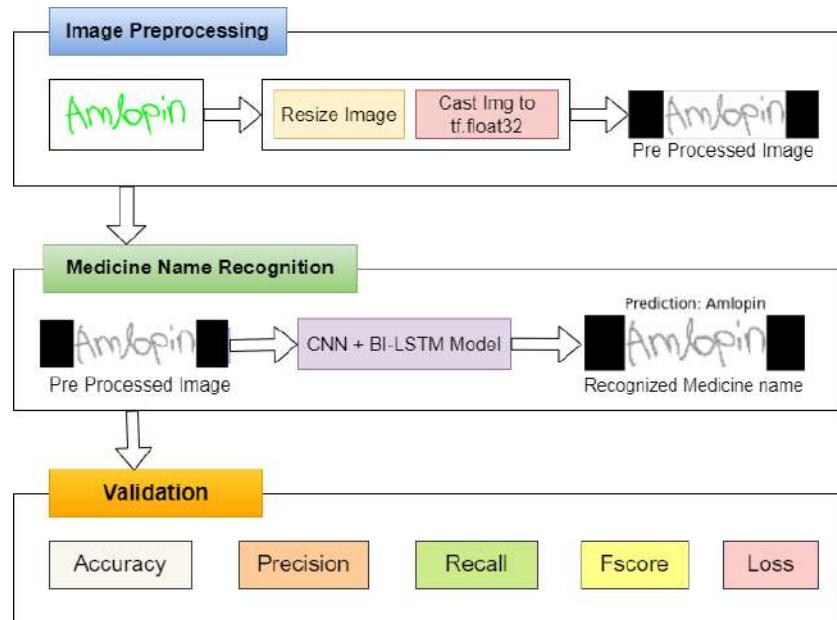


Fig.2. Architecture design for our work

The Bi LSTM layer aids in decoding of the convolution layer-generated feature mapping. Next create a layer for label input for the appropriate images, followed by a thick layer. To find the CTC loss, the last layer would be the CTC layer. The next step is to determine the callback distance. During this stage, ctc decoding will be done using predictions. Then, using `tf.sparse.from_dense` and `dtype` as an `int64`, sparse the predictions made from dense. Identify a point when an increase in epoch values leads to a rise in loss value and the model will no longer improve performance at which point training may be stopped. Next, build the model and train it with various rising epoch values, setting checkpoints as necessary. The count variable can be used to determine the model's accuracy by passing some data as input, checking the total count of right predictions, and calculating the percentage of right predictions. The accuracy will be proportional to the initial weights which are being established in the input layer. Setting appropriate starting weight values for the input layer and subsequent layers, as well as choosing an appropriate activation function, are required. Relu inside a convolution layer and Dense layer are typically utilised as activation functions for models like CNN and LSTMs.

StringLookup Layer:

This is proposed preprocessing layer which is being used. It converts characters in each word while training into integers. It converts each character in a vocabulary to an integer. StringLookup Layer converts integers to characters during prediction. During the conversion of num back to character the invert should be set as True. If the training label is ‘crocin’ then the vocabulary will be {c,r,o,c,i,n}.

CNN + Bi LSTM:

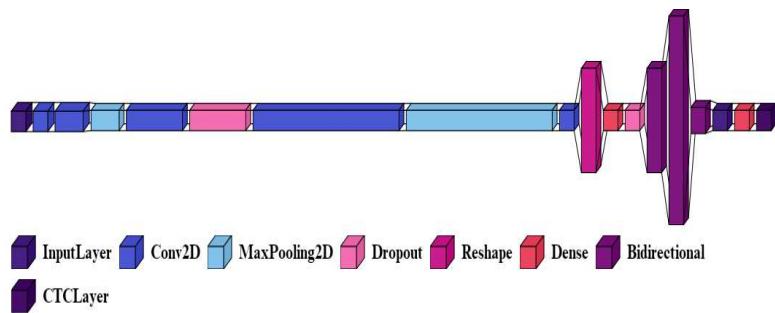


Fig.3. Model topology for Bi LSTM

The fig(3) shows model design for Bi LSTM Model. The Bidirectional layer in fig(3) represents the Bi LSTM Layer. The initial layer is input layer of height 32, width 128 and the channel is 1 because it's a grayscale image. The proposed model with Bi-LSTM contains five convolutional layers with filters 32,128,256,1024 and 64. For three Bi LSTM layers number of hidden cells are 512,1024,64. The CTC layer computes the character wise error rate instead of word wise error rate and returns loss value for every step per epoch.

CTC LAYER:

The proposed CTC layer merges repeated related characters. The first phase in this layer is predicting the tokens in a sequential order. Second phase is to merge repetitive characters and drop noisy tokens. CTC Layer provides final output after phases 1 and 2. This Layer offers a Loss function to compare anticipated and actual values.

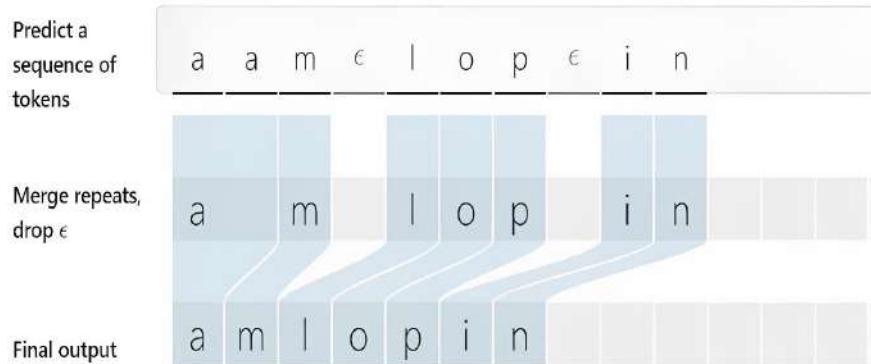


Fig. 4. Functionality of CTC Layer

Fig(4) shows CTC Layer's operation. It doesn't use any traditional aligning methods, instead it will eliminate the process of Alignment. Bi-LSTM or Bi-GRU may output 'Amlopin' as a series of characters. The custom CTC Layer filters characters in each word. In the proposed network design, CTC Layer input and output shapes are (None,32,81) itself. Every training step calculates CTC loss. CTC batch cost's arguments are y pred, y true, label length, and input length. It returns each element's loss. Image input determines this argument's value.

CNN + Bi-Directional-GRU:

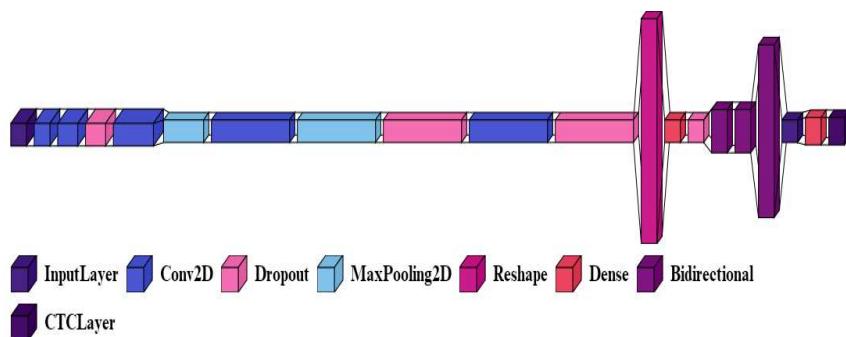


Fig. 5. Model Architecture for Bi-Directional GRU

Fig(5) demonstrates Bi-Directional GRU model building. Input layer shape is (128,32,1) with width, height, and channel. Dropout layers between convolutional and pooling layers prevent overfitting. The Dropout value used is 0.25. 5 convolution layers with filter sizes 32,128,256,512,1024. The next 3 Bi-Directional

GRU layers have 128,128,512 hidden cells. The CTC LAYER works similarly to the prior Bi-LSTM model fig (4).

After comparing the both models, Bi-LSTM has more training parameters than Bi-GRU. The model's hidden units determine the total number of trainable parameters. The total trainable parameters are 6,907,025. Consider character error rate for an effective model.

3.4 Selection of Algorithm

The implementation of proposed model is done with two different algorithms, they are Bi- LSTM and Bi-GRU. The CNN is common in both models. But the type of RNN used is different. The model uses algorithms of CNN + Bi-Directional LSTM[1] in first model and CNN + Bi-Directional GRU[17] in second model. Here, the comparison is done in order to determine which model is performing well. From fig(2) in medicine name recognition phase the algorithms used is different, but other steps remain same. Both LSTM and GRU are Recurrent neural networks but the difference is the size of data they can handle and gates present. Using IAM dataset, the training dataset size is 86,800 images. Bi-Directional LSTM is excellent at handling huge data, followed by GRU. GRU is preferred for small datasets. Input, Output, Forget are LSTM gates. GRU only has update and reset gates. GRU is simpler than LSTM. So, its better to use GRU for small sized data and LSTM for large sized data. The selection of this two algorithms is done in order to test which algorithm will perform well.

The Mathematical Notation of Bi-Directional LSTM as follows:

$$\hat{p}^{(t)} = g(W_x[\vec{a}^{<t>}, \tilde{a}^{<t>}] + b_x) \quad (1)$$

From eq.(1), \hat{p} is used for representation of output, $\hat{p}^{(t)}$ will represent output at t^{th} unit of time. The g represents the hidden layer function. W_x denotes the hidden layer weights matrix. And similarly b_x denotes hidden layer vector for bias values. $\vec{a}^{<t>}$ gives the forward hidden sequence and $\tilde{a}^{<t>}$ gives the backward sequence. As the employment of Bi-directional layers is being done, the two directional sequences can be obtained. Here, the iterations are done on forward and backward sequences and then the output will get updated when concatenation is done for each word in forward and backward sequence.

The Mathematical Notation of Bi-Directional GRU as follows:

$$h_t = G_{\text{forward}}(x_t, \overrightarrow{h_{t-1}}) \oplus G_{\text{backward}}(x_t, \overleftarrow{h_{t+1}}) \quad (2)$$

Eq.(2) Similar to Bi-Directional LSTM, the Bi-Directional GRU also allows data sequence in two directions forward and backward. From eq.(2), $G_{forward}$ is a GRU function that denotes the data sequence flow in Forward direction and $G_{backward}$ is a GRU function that denotes the data sequence flow in Backward direction. \oplus is vector concatenation operator for $G_{forward}$ and $G_{backward}$. Data sequence flows. $G_{forward}(x_t, \overrightarrow{h_{t-1}})$ denotes forward GRU's state and $G_{backward}(x_t, \overleftarrow{h_{t+1}})$ denotes backward GRU's state. x_t denotes input vector. h_t denotes output of cell at time t. By performing concatenation between forward and backward GRU states gives the output h_t . From the eq.(2), the concatenation of both forward and backward sequences makes the model to access previous states as well.

4 Result and Observations

4.1 Test case Results

Testcase results are below. Model predicts based on custom image inputs.

Predictions:

'grey', 'what', 'earns', 'with']

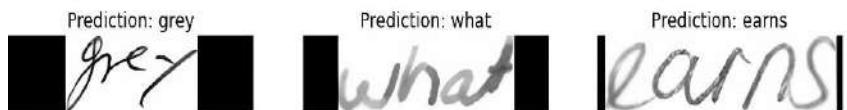


Fig. 6. Output Test Case predicted by the model

Fig(6) displays IAM Testing test case results. Fig(6) shows the model's predictions after 25 Epochs of training. All three testing dataset images were successfully predicted by the model. First, the image was preprocessed and passed to the model for prediction.

4.2 Observations and Analysis

Accuracy observations for Bi-Directional LSTM and Bi-Directional GRU at different epochs are shown below.

Table. 1. Comparison of Bi-LSTM and Bi-GRU Accuracies

Epochs	Accuracy (%)	
	BI-LSTM	BI-GRU
20	75	69
25	78	74
30	81	77

From table(1), Two models (one with Bi-LSTM and another with Bi-GRU) are trained and tested to compare accuracy. At 30th Epoch, Bi-Directional LSTM had 81% Accuracy and Bi-Directional GRU 77%. Bi-Directional LSTM Algorithm training took 5 hours and Bi-Directional GRU training took 3 hours. Bi-Directional GRU, a less sophisticated model, trained 30 epochs in less time than Bi-Directional LSTM, but had lower accuracy. Bi-Directional LSTMs outperform Bi-Directional GRUs on larger datasets.

Bi-Directional GRU has 88% training data while Bi-Directional LSTM has 90%. To avoid overfitting, control validation loss. By adding enough dropout value, overfitting can be controlled, and data quantity can affect validation loss.

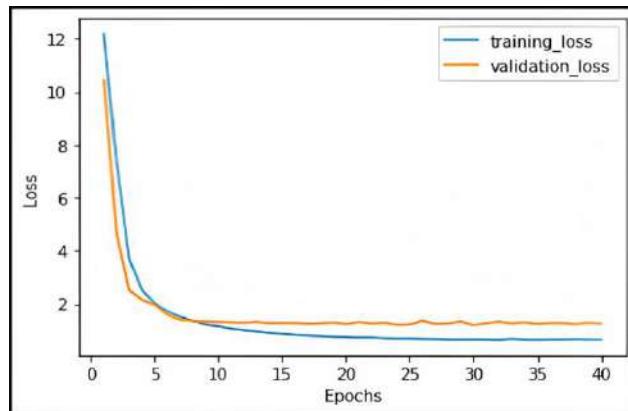


Fig. 7. Loss vs Epochs with Bi-Directional LSTM

Fig(7) plots Epochs versus Loss. Loss values are achieved after 40 epochs of training and saving the best weights with checkpoints. Training and validation loss are minimal at 30th epoch. The training loss is 0.6 and validation loss is 1.24. Training loss and validation loss are close together on the graph, indicating no overfitting. If validation loss increases while training loss doesn't, that's model overfitting. Adding a dropout layer helps to avoid overfitting. The number of neurons to be dropped from each hidden layer is dropout percentage. In proposed architecture all the dropout layers are having 25% of Dropout.

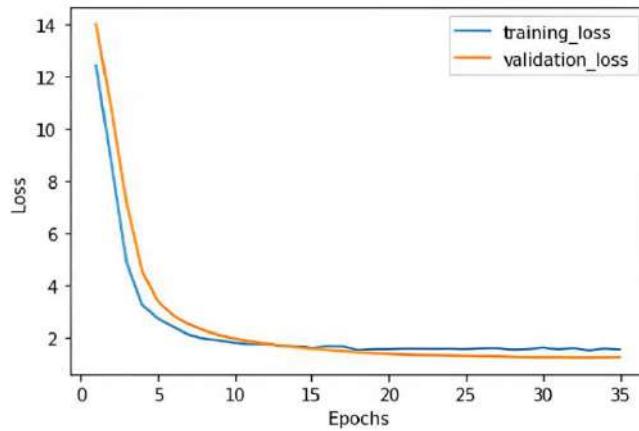


Fig. 8. Loss vs Epochs with Bi-Directional GRU

Fig(8) shows Bi-Directional GRU loss versus epochs graph. Model was trained for 35 epochs and best weights are restored. At 30th epoch model has minimal training and validation loss. Bi-Directional GRU has 1.5 and 1.3 training and validation losses respectively. Bi-Directional LSTM model has 81% accuracy, 0.96 precision, 0.83 recall, and 0.89 F-score. Bi-Directional GRU has 77% accuracy, 0.95 precision, 0.80 recall, and 0.86 F-score.

This Metric shows that Bi-Directional LSTM performs well with large training and validation datasets. As LSTMs have more gates than GRUs, the suggested method is more complex but can perform better than alternatives with fewer gates. For realtime use case, maintain quality of dataset used for training model. The current handwriting recognition model should be trained with 50 Doctors' handwriting styles. This lets the model detect any handwriting and make more accurate predictions. Training and validation datasets should have diverse calligraphy styles. This makes the model more resilient so it can properly predict most inputs.

5 Conclusion and Future Work

This model helps Pharmacists and normal people to recognize the medicine name accurately which is present in the Doctor's Handwritten Prescription. This effectively offers text in all handwriting typefaces. The feature extraction is carried out by Convolutional Neural Networks with many layers[19], and the decoding of the extracted features into English letters is assisted by Bi-LSTMs. The CTC is employed to circumvent the fact that the true alignment between the input and the output is unknown. To accurately identify language specific to prescriptions provided by the doctors, more bias is applied to words that are present in a manually

produced corpus. More data increases model accuracy. Compared to Bi-Directional GRU[23], Bi-Directional LSTMs perform better and are more accurate with large datasets..

Future work: Training and validation dataset size can be increased and diverse handwriting images can be used for training in order to improve model's recognizing ability. Advanced algorithms can boost model performance. Pre trained models can be used which may boost model's performance.

References

- [1] T. Jain, R. Sharma and R. Malhotra, "Handwriting Recognition for Medical Prescriptions using a CNN-Bi-LSTM Model," 2021 6th International Conference for Convergence in Technology (I2CT), 2021, pp. 1-4. (T. Jain, 2021)
- [2] S. Tabassum et al., "Recognition of Doctors' Cursive Handwritten Medical Words by using Bidirectional LSTM and SRP Data Augmentation," 2021 IEEE Technology & Engineering Management Conference - Europe (TEMSCON-EUR), 2021, pp. 1-6.
- [3] L. J. Fajardo et al, "Doctor's Cursive Handwriting Recognition System Using Deep Learning," 2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2019, pp. 1-6.
- [4] Espana-Boquera,S., Castro-Bleda, M.J."A Spanish dataset for reproducible benchmarked offline handwriting recognition". Lang Resources & Evaluation 56, 1009–1022 (2022).
- [5] Maalej,Kherallah, M. "New MDLSTM-based designs with data augmentation for offline Arabic handwriting recognition". Multimed Tools Appl 81, 10243–10260 (2022).
- [6] A. Harikrishnan, S. Sethi and R. Pandey,"Handwritten Digit Recognition with Feed-Forward Multi-Layer Perceptron and Convolutional Neural Network Architectures", 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 2020, pp. 398-402.
- [7] A. Nikitha, J. Geetha and D. S. JayaLakshmi, "Handwritten Text Recognition using Deep Learning," 2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (2020).
- [8] M. I. Swindall et al., "Exploring Learning Approaches for Ancient Greek Character Recognition with Citizen Science Data," 2021 IEEE 17th International Conference on eScience (eScience), 2021.
- [9] S. Hassan,A. Irfan,A. Mirza and I. Siddiqi, "Cursive Handwritten Text Recognition using Bi-Directional LSTMs: A Case Study on Urdu Handwriting," 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), 2019.
- [10] ul Sehr Zia, N., Naeem, M.F., Raza, S.M.K. et al."A convolutional recursive deep architecture for unconstrained Urdu handwriting recognition". Neural Comput &

Applic 34, 1635–1648 (2022).

- [11] Alrobah, N., Albahli, S."Arabic Handwritten Recognition Using Deep Learning: A Survey". Arab J Sci Eng 47, 9943–9963 (2022).
- [12] U. Shaw, Tania, R. Mamgai and I. Malhotra, "Medical Handwritten Prescription Recognition and Information Retrieval using Neural Network," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), 2021, pp. 46-50
- [13] S. Sharma and S. Gupta, "Recognition of Various Scripts Using Machine Learning and Deep Learning techniques-A Review," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC 2021) pp. 84-89.
- [14] Sethy, A., Patra, P.K., Nayak, S.R. "A Deep Convolutional Neural Network-Based Approach for Handwritten Recognition System". In: Das, A.K., Nayak, J., Naik, B., Dutta, S., Pelusi, D. (eds) Computational Intelligence in Pattern Recognition . Advances in Intelligent Systems and Computing, vol 1349. Springer, Singapore (2022).
- [15] Altwaijry, N., Al-Turaiki, I. Arabic handwriting recognition system using convolutional neural network. Neural Comput & Applic 33, 2249–2261 (2021).
- [16] Gupta, D., & Bag, S. (2020). "CNN-based multilingual handwritten numeral recognition: A fusion-free approach". Expert Systems with Applications, pp 1-14
- [17] S. Haboubi, T. Guesmi, B. M. Alshammari, K. Alqunun, A. S. Alshammari *et al.*, "Improving cnn-bgru hybrid network for arabic handwritten text recognition," Computers, Materials & Continua, vol. 73, no.3, pp. 5385–5397, (2022).
- [18] H. Lamtougui, H. E. Mouftahij, H. Fouadi and K. Satori, "An efficient hybrid model for arabic text recognition," *Computers, Materials & Continua*, vol. 74, no.2, pp. 2871–2888, (2022).
- [19] G. Ahmed, T. Alyas, M. Waseem Iqbal, M. Usman Ashraf, A. Mohammed Alghamdi *et al.*, "Recognition of urdu handwritten alphabet using convolutional neural network (cnn)," *Computers, Materials & Continua*, vol. 73, no.2, pp. 2967–2984, (2022).
- [20] K. O. Mohammed Aarif and P. Sivakumar, "Multi-domain deep convolutional neural network for ancient urdu text recognition system," *Intelligent Automation & Soft Computing*, vol. 33, no.1, pp. 275–289, (2022).
- [21] S.Abbas, Y. Alhwaiti, A. Fatima, M. A. Khan, M. Adnan Khan *et al.*, "Convolutional neural network based intelligent handwritten document recognition," *Computers, Materials & Continua*, vol. 70, no.3, pp. 4563–4581, (2022).
- [22] Y. Xue, Y. Tong, Z. Yuan, S. Su, A. Slowik *et al.*, "Handwritten character recognition based on improved convolutional neural network," *Intelligent Automation & Soft Computing*, vol. 29, no.2, pp. 497–509,(2021).
- [23] A. Abdallah, M. Hamada, and D. Nurseitov, "Attention-Based Fully Gated CNN-BGRU for Russian Handwritten Text," *Journal of Imaging*, vol. 6, no. 12, p. 141(2020).

Abductive Inference of Conclusions with Check of Additional Premises Literals Correctness Interpretation

*Vasily Meltsov¹, Dmitry Strabykin¹, and Alexander Krutikov²

¹Department of Electronic Computers, Vyatka State University, Russia

¹*meltsov69@mail.ru

²Institute of Mathematics and Information Systems, Vyatka State University,

Russia

²usr09603@vyatsu.ru

Abstract. At the present stage of the development of computer technology and information technology, the theory and methods of reasoning modeling play an important role in creating artificial intelligence systems. Reasoning modeling can be used, for example, in managing the functioning of complex intelligent information and control systems in order to explain and justify the decisions they recommend. The article considers a special method of abductive logical inference with checking the correctness of the interpretation of literals of additional premises. The method under consideration, in addition to explaining the course of inference with the help of schemes, allows you to check the correctness of the interpretation of new "facts" formed in the process of inference, taking into account the ontology of a particular subject area. The main advantage of the proposed method is the parallel execution of disjunctive division operations in the inference procedure.

Keywords: Knowledge processing, intelligent systems, abductive inference, predicate calculus, additional premises.

1 Introduction

The rapid development of information technology and computer technology in recent years has allowed the transition to the creation of high-performance artificial intelligence systems [1]. The use of such systems is particularly effective in solving problems in such areas as business management [2], medical and technical diagnostics [3], transportation logistics, image processing, logical forecasting, semantic analysis of texts, etc. Along with the successful application of artificial neural networks and various machine learning algorithms [4], theory and methods of reasoning modeling play an important role in the design of high-performance knowledge processing systems [5], [6].

Reasoning modeling can be used, for example, in monitoring the functioning of complex intelligent information and control systems in order to explain and justify the solutions recommended by them [7]. Representation of knowledge in

the form of logical formulas and deductive logical inference allow to guarantee the validity of the recommended solutions, provided the validity of the initial statements (premises).

In the process of deductive logical inference, new statements are formed as consequences of already existing statements, and their truthfulness is conditioned by the truthfulness of previous statements [8]. To simplify the explanation and justification of the recommended solution, the process of logical deduction is visualized in the form of a special inference scheme, reflecting the used statements and their relationship, and allowing to analyze the course of reasoning, generated by logical inference [9].

When intelligent information and control systems operate under conditions of incomplete or incorrect information, it is of interest to model plausible reasoning to explain and justify the recommended solutions [10]. In this case methods of abductive logical inference can be used, allowing to determine the missing secondary premises and obtain a successful deductive logical inference [11].

Deductive and abductive logical inference can be explained by the following example (Table 1) [12].

Table 1. An example of a logical task.

No	Initial premises	Interpretation	In predicate logic
1	Rule (main premise)	[All people are mortal]	$\forall(x)\text{HUMAN}(x)\Rightarrow\text{MORTAL}(x)$.
2	Fact (minor premise)	[Socrates is a man]	$\text{HUMAN}(\text{SOCRAT})$.
3	Purpose (conclusion)	[Socrates is mortal]	$\text{MORTAL}(\text{SOCRAT})$.

Deduction is the derivation of goal 3 from premises 1 and 2. Abduction in the above example would mean deriving premise 2 from premise 1 and purpose 3.

In general, the process of abductive inference may fail because it does not allow obtaining a formula for an additional premise or the obtained additional premise will contradict the existing main premise. At the same time, even if the formula of an additional premise does not lead to inconsistency in the knowledge base, it can contain literals that have no correct interpretation in the subject area and are therefore unacceptable.

Obviously, the presence of unacceptable literals in the additional premises can be determined by a user of intelligent information control system when analyzing the literals for the possibility of their interpretation in the subject area [13]. However, this approach has serious drawbacks: it slows down the process of controlling the functioning of an intelligent information-management system and risks the appearance of user errors.

In this connection, it is of interest to automatically check all literals of additional premises in order to correctly interpret them in the subject area on the basis of a special knowledge base containing unacceptable facts and rules.

2 Method of abductive logical inference

2.1 Formal problem statement

The task of abductive logical inference can be formulated as follows. There are initial consistent premises given as a set of disjuncts $M = \{D_1, D_2, \dots, D_K\}$. There is also a conclusion represented by the set of deducible disjuncts $m = \{d_1, \dots, d_T\}$. If the deductive inference is not successful, then it is necessary to: define the set of additional premises MD , addition of which to the set of initial premises M ensures a successful deductive inference: $M' \Rightarrow dt$, где $M' = M \cup MD$, $t=1, \dots, T$. The new set of premises M' should be consistent.

2.2 Method of abductive inference

The method of special abductive inference of the conclusion is based on the abductive inference procedure (w -procedure) and the procedures of full (Ω -procedure) and partial division of disjuncts (ω -procedure) [11], addition of which to the set of initial premises M ensures a successful deductive inference: $M' \Rightarrow dt$, где $M' = M \cup MD$, $t=1, \dots, T$. The new set of premises M' should be consistent.

Partial disjuncts division. Partial disjunct division is performed using a special procedure:

$$\omega = \langle b, d, q, n \rangle, \text{ where:}$$

b – remainder-dividend (disjunct of premise) used to obtain the remainders;

d – remainder-divisor (disjunct of conclusion) involved in the formation of remainders;

q – is a particular attribute of the solution having three values: "0" - at least one zero remainder is obtained, "1" - all obtained remainders are equal to one, "g" - more than one remainder unequal to one is obtained in the absence of zero remainders;

$n = \{ \langle bt, dt \rangle, t=1, \dots, T \}$ – the set of pairs consisting of the new remainder-dividend bt and its corresponding remainder-divisor dt .

In the process of disjunct division the literals that coincide after unification with the literals of the disjunct-divisor are excluded from the disjunct-divisor. If

the last literal is excluded from the disjunct-divisible, the remainder is “0”, if the literals are not unified, the remainder is “1”. In other cases, the remainder consists of the literals of the disjunct-divident obtained after applying a unifying substitution to them and excluding the matching literal. Since a unifying substitution can also change the variable values in the literals of the disjunct-divisor, each new residual corresponds to a new modified version of the disjunct-divisor. A detailed description of the partial disjuncts division procedure can be found in [11].

Complete disjunct division. Complete disjunct division is aimed at obtaining so called finite remainders. A remainder-dividend b is finite for a remainder-divisor d if applying the procedure ω to it does not produce new remainders other than “1”. A finite remainder b or a new remainder b_t can be a consummate remainder. When disjuncts are completely divided, the resulting set of remainders includes principal remainders, which include uniliterate final remainders as well as final remainders with two or more nonvariable literals.

The complete division of disjunct D into disjunct d is performed considering the facts (single-literal initial disjuncts) with the help of a special procedure:

$$\Omega = \langle D, d, Q, N \rangle, \text{ where:}$$

Q – the solution flag having three values: "0" - the solution is found, "1" - the disjunct D has no d remainders other than unity, "G" - the set of finite remainders is obtained;

N – set consisting of finite remainders E_j ($N = \{E_j, j=1, \dots, J\}$). At $Q=0$ $N=\{0\}$, and at $Q=1$ $N=\{1\}$.

The formation of a set of residuals is done by repeated application of ω -procedures and consists of a series of steps. At each step, ω -procedures are applied to the existing divisible and divisible residuals, forming new divisible residuals and new divisible residuals that are used as input in the next step. The process ends when the next step reveals a ω -procedure that has produced zero residuals ($q=0$), or all ω -procedures of this step have produced traits that indicate the final residuals ($q=1$).

The abductive inference procedure. The procedure allows you to make a step of inference, converting the inferred disjunct into a set of new disjuncts needed to continue inference in the next step, and also calculate an auxiliary disjunct that is used to form additional premises. We define the abductive inference procedure w as:

$$w = \langle M, d, o, q, p, m, \partial \rangle, \text{ where:}$$

$M = \{D_1, D_2, \dots, D_i, \dots, D_l\}$ – set of initial disjuncts (initial premises);

$d = L_1 \vee L_2 \vee \dots \vee L_k \vee \dots \vee L_K$ – inferred disjunct (conclusion);

$o = \langle c, C \rangle$ – a pair of current remainder sets consisting of the sets of residuals formed before (c) and after (C) procedure w;

q – attribute of the solution, which has two values: 0 - there is a solution and 1 - no solution;

p – attribute of the inference termination, which has two values: 0 - continuation of the inference is possible and 1 - continuation of the inference is impossible;

$m = \{d_g, g=1, \dots, G\}$ – is the set of new inferred disjuncts d_g . If $p=1$ $m=\emptyset$;

∂ - is the complement, which is an auxiliary remainder used in forming disjuncts of additional initial premises. The augment is computed by means of partial disjunct group division ϖ operation which is performed by partial division of disjunct d into a subset of initial disjuncts M_0 ($M_0 \subseteq M$) that have nonunit remains after division by disjunct d. At the first step of the inference for $d \varpi M_0 = 1$ we take $\partial = 1$, and at later steps $\partial = d$.

The inference procedure uses the previously discussed complete disjunction division Ω -procedure as a subprocedure.

The abductive inference procedure is applicable if $M \neq \emptyset$ and $J_i, K \geq 1$ ($i=1, \dots, I$), otherwise the attribute $p=1$ is immediately set. When the procedure is used for the first time, the input set of current remainders includes inversions of literals of the conclusion d: $c = \{\neg L_k, k=1, \dots, K\}$. In the inference procedure the following actions are performed.

1. Remainders of initial disjuncts are formed:

- disjuncts of initial premises are divided by Ω -procedure into conclusion: $D_i \Omega d = \bigwedge_{r=1}^{R_i} b_{ir}$, $i=1, \dots, I$. And if at least one remainder $b_{ir} = 0$ (at least in one Ω -procedure $Q=0$), then $q=0$ is taken and proceeds to step 4, otherwise the following action is performed.
- residuals $b_{ir} = 1$ are excluded. If all remainders are equal to 1 (in all Ω -procedures $Q=1$), no inference is possible; take $p=1$ and proceed to step 4, otherwise the following action is performed.
- the set E of remainders of initial disjuncts is formed: $E = \{b_s, s=1, \dots, S\}$, which includes only remainders b_{ir} that differ from each other. Remainders that coincide with or are absorbed by the remainders of c are excluded from this set. This results in a set e of remainders of the original sequences that does not contain the previously obtained remainders. If $e = \emptyset$, no inference is possible; $p = 1$ is taken and we proceed to step 4, otherwise the next step is executed.

2. The conjunctive condition of the inference termination is checked. The output set of current remainders is formed: $C = c \cup e$, $C = \{b_n, n=1, \dots, N\}$. An expression $\bigwedge_{n=1}^N b_n$ composed, which is simplified by multiplying the residuals and excluding conjunctions containing $L \neg L$ (0L) type factors. If the expression thus transformed equals zero, the inference is successfully completed ($q=0$) and proceeds to step 4; otherwise the next item is executed.

3. New disjuncts are formed. The expression, $\bigwedge_{v=1}^V b_v$, is left from the remainders of the set $e=\{bv, v=1, \dots, V\}$, which is simplified by multiplying the remainders and excluding conjunctions that contain $L \bar{L}$ (OL) type doublers. The expression obtained after multiplication of disjuncts and simplification is a disjunctive form of the form: $X_1 \vee X_2 \vee \dots \vee X_g \vee \dots \vee X_H$, where $X_g = L_1^g L_2^g \dots L_{H_g}^g$. Inversions of conjunctions X_g form new derivable disjuncts $\neg L_1^g \vee \neg L_2^g \vee \dots \vee \neg L_{H_g}^g$, $g=1, 2, \dots, G$. New conclusions are included in the set m . Attribute of the inference termination $p=0$ is set.

4. End of procedure.

Method of abductive inference. The inference is represented by a set of disjuncts. Logical inference is reduced to multiple application of w -procedures and consists of a number of steps. At each step of inference w procedures are applied to existing inferred and initial disjuncts forming new inferred disjuncts that are used at the next step. Besides on each step with help of w -procedures add-ons are calculated which serve as a basis for formulae of Ψ additional premises used for forming of set MD of additional assumption disjuncts. In the process of logical inference two flags are formed at the end of the step:

Q - solution general flag ($Q=0$ - there are solutions, $Q=1$ - no solutions);

P - inference continuation general flag ($P=0$ - inference continuation is possible, $P=1$ - inference continuation is impossible). The inference process ends when $Q=0$ (no additional premises are required) or $P=1$.

The inference process can also be terminated when a given number of inference steps is reached. The method is applicable if the inferred disjuncts are not tautologies and their conjunction is a contradiction.

For a more complete description of the method we will use a special index function that provides unique identification of each procedure and its parameters at the h -th step of the inference ($h=1, \dots, H$).

The index function $i(h)$ is defined for the index variable t inductively as follows:

- $i(1)=t, t=1, \dots, T;$
- $i(2)=t.t_i=i(1).t_{i(1)}, t_{i(1)}=1, \dots, T_{i(1)};$
- $i(3)=t.t_i.t_E (E=t.t_i), i(3)=i(2).t_{i(2)}, t_{i(2)}=1, \dots, T_{i(2)};$
- etc.

In general case: $i(h+1)=i(h).t_{i(h)}$, $t_{i(h)}=1, \dots, T_{i(h)}$. Let us assume that $i(0)$ means that there is no index for the indexed variable, e.g. $T_{i(0)}=T$, and that $i(1)=i(0).t_{i(0)}=t$. Then the description of the method can be represented as follows.

1. Initial values are defined: $h=1$, $h_{\max}=H$, $M=\{D_1, \dots, D_K\}$, $m=\{d_1, \dots, d_T\}$, $i(1)=t; t=1, \dots, T$, where T is the number of inferred disjuncts; $o=<c, C>$, $c=C=\emptyset$.

2. The w -procedures of the current (h -th) step are executed. For derived disjuncts of set m on the first step ($h=1$) and on the next step ($h>1$) for disjuncts of

sets $m_{i(h-1)}$, obtained in the procedures of the previous step, characterized by features $p_{i(h-1)}=0$ and $q_{i(h-1)}=1$, the following w -procedures are executed:

$$w_{i(h)} = \langle M, d_{i(h)}, o_{i(h)}, q_{i(h)}, p_{i(h)}, m_{i(h)}, \hat{d}_{i(h)} \rangle, i(h) = i(h-1).t_{i(h-1)}, t_{i(h-1)} = 1, \dots, T_{i(h-1)}.$$

3. The solution general flag (Q^h) and the inference continuation general flag (P^h) are formed, the formula (Ψ) of the additional premises is constructed:

$$\begin{aligned} Q^h &= \bigvee_{A=1}^B q_{i(h)} \vee S_{i(h-1)}, P^h = \bigwedge_{A=1}^B (\neg q_{i(h)} \vee p_{i(h)}) \vee S_{i(h-1)}; \\ \Psi_{i(h-1)} &= \neg S_{i(h-1)} \wedge_{A=1}^B (\hat{d}_{i(h)} \vee \neg p_{i(h)} \Psi_{i(h)} \vee \neg q_{i(h)}) \vee S_{i(h-1)} \wedge_{A=1}^B d_{i(h)}; \end{aligned}$$

where:

$$A = t_{i(h-1)} \text{ and } B = T_{i(h-1)};$$

$S_{i(h-1)}$ is a flag of consistency of values of common variables in subsets $m_{i(h-1)}$ of derived disjuncts $d_{i(h)}$, which these variables took after performing procedures $w_{i(h)}$ ($S_{i(h-1)}=0$ - values are consistent or disjuncts have no common variables, $S_{i(h-1)}=1$ - values of common variables are not consistent).

4. The values of the general signs of the solution (Q^h) and the inference continuation general flag (P^h) are checked. If $Q^h=0$, the inference is completed successfully. In this case $Q=0$, $P=P^h$, $M^D=\emptyset$ are set and the next step is executed. If $Q^h=1$, the sign P^h is analyzed. If $P^h=0$, then at $h < h_{\max}$ h is increased by one and executed step 2 (the process continues). If $P^h=1$ or $h=h_{\max}$, then $P=1$ is set and the next step is executed.

5. The inference process is completed. Moreover, at $Q=0$ the inference is completed successfully and no additional premises are required ($M^D=\emptyset$). At $Q=1$ (no solution) the inference is unsuccessful, but as a result of the inference the formula of additional premises Ψ is constructed. If $\Psi \neq 1$ and $\Psi \neq 0$ then the set of disjuncts of additional premises MD is formed. Additional disjuncts are obtained by converting the formula of additional propositions into conjunctive normative form:

$$\Psi = D_{K+1} D_{K+2} \dots D_{K+n} \dots D_{K+N}, M^D = \{D_{K+n}, n=1, \dots, N\}.$$

6. The literals of additional premises are checked for the correctness of interpretation in the domain. The literals representing unacceptable statements according to the conditions of the problem are excluded from the disjuncts of additional premises of the MD set. To determine the correctness of interpretation of literals of additional premises in the domain, one can use a special knowledge base containing unacceptable facts and rules. The logical following of literal of an additional premise from the facts of the specified knowledge base will mean its unacceptability.

2.2 Example of abductive inference

As an example of abductive logical inference, consider solving the following problem (Table 2) [14]. Initially the knowledge base contains 3 facts and 2 rules-disjuncts.

Table 2. The set of initial premises and conclusion

D	Knowledge	Premises/conclusion	Clauses (disjuncts)
D ₁	Fact 1	directs(Serge,Boris)	D1=P(c,b)
D ₂	Fact 2	directs(Boris, Anna)	D2=P(b,a)
D ₃	Fact 3	directs(Anna, Alex)	D3=P(a,e)
D ₄	Rule 1	directs(x,y)→reports(y,x)	D4=¬P(x,y)∨O(y,x)
D ₅	Rule 2	directs(z,u)&reports(s,u)→ reports(s,z)	D5=¬P(z,u)∨¬O(s,u)∨O(s,z)
r	Conclusion	"Does Alex report to Serge?"	d=O(e,c)?

For the given initial data the problem is solved by means of deductive logical inference, which ends successfully: the conclusion is a consequence of the premises. The scheme of logical deduction of the conclusion is shown in Fig. 1.

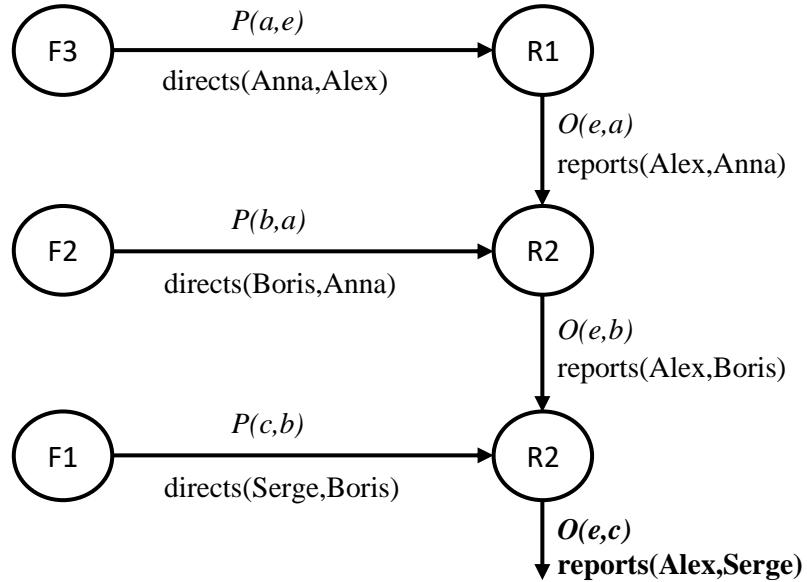


Fig. 1. Scheme of inference to conclusion «reports(Alex,Serge)»

Let us exclude fact 2 and its corresponding disjunct $D_2=P(b,a)$ from the original data. After excluding this fact, deductive logical inference becomes impossible.

The construction of abductive explanations, which is carried out by abductive logical inference, makes it possible to ensure successful deductive inference.

Abductive logical inference in this example consists of three steps ($h=1,2,3$) where one deduction procedure (w_1 и $w_{1.1}$) is executed on the first and second steps and two procedures on the third step ($w_{1.1.1}$ и $w_{1.1.2}$). Table 3 summarizes the main results of the procedures and the formulae of additional premises by steps. At the third step of derivation after matching the values of common variable u_2 the formula for additional premises takes the following form: $\Psi=P(c,e) \vee P(b,a)$.

Thus, in the process of abductive inference of the intermediate conclusion $d_1=O(e,c)$, we obtain a set of additional premises $M^D=\{D_6\}$, where $D_6=P(c,e) \vee P(b,a)$ is formed by performing the operation of disjunct division in three inference's steps.

Table 3. The process of abductive logical inference

<i>h</i>	<i>w</i>	<i>d</i>	<i>q</i>	<i>p</i>	<i>m</i>	<i>δ</i>	<i>Q</i>	<i>P</i>	<i>S</i>	Ψ
1	$w_1=<M,d_1,o_1,$ $q_1,p_1,m_1,\delta_1>$	$O(e,c)$	1	0	$\{d_1\}$	0	1	0	0	Ψ_1
2	$w_{1.1}=<M,d_{1.1},o_{1.1},$ $q_{1.1},p_{1.1},m_{1.1},\delta_{1.1}>$	$P(c,e) \vee O(e,b)$	1	0	$\frac{\{d_{1.1.1}\}}{d_{1.1.2}}$	$P(c,e)$	1	0	0	$P(c,e) \vee \Psi_{1.1}$
3	$w_{1.1.1}=<M,d_{1.1.1},o_{1.1.1},$ $q_{1.1.1},p_{1.1.1},m_{1.1.1},\delta_{1.1.1}>$	$P(b,u_2)$	1	1	\emptyset	$P(b,u_2)$				
	$w_{1.1.2}=<M,d_{1.1.2},o_{1.1.2},$ $q_{1.1.2},p_{1.1.2},m_{1.1.2},\delta_{1.1.2}>$	$O(e,u_2)$	0	1	\emptyset	0	1	1	0	$P(c,e) \vee P(b,a)$

The resulting complementary premise does not contradict the main premises of D_1-D_5 . However, it is necessary to check the literals of the additional premise for the correctness of interpretation in the domain [15]. The literals representing unacceptable statements according to the conditions of the problem should be excluded from the disjunct of the additional premise [16].

3. Checking the correctness of literal interpretation

In principle, the user of the intelligent information and control system himself can carry out the necessary chec. In the example, the analysis of the literals of the supplementary premise shows that the literal $P(c,e)$ - "directs(Serge,Alex)" cannot be used in the supplementary premise based on the meaning of the problem,

because the base contains the fact $D3=P(a,e)$ - Alex is already under the direct control of Anna (*two direct leaders of one subordinate are not allowed*). However, it is more effective to check the literals of the additional premise for the correct interpretation in the subject area can be carried out on the basis of a special knowledge base in automatic mode. Then the successful logical conclusion of the additional premise literal from this special knowledge base will mean its unacceptability.

In the example under consideration for formal determination of correctness of interpretation of literals of an additional premise in the subject domain can be used special knowledge base, containing the following premises: $D*_1=P(c,b)$; $D*_2=P(a,e)$; $D*_3=\neg P(u,w) \vee P(v,w)$ (the presence of two direct managers for one subordinate). Deductive logical inference of literals of the additional premise $P(c,e)$ and $P(b,a)$ from the special knowledge base $D*_1, D*_2, D*_3$ is performed by means of partial (ω) and complete (Ω) disjunct division procedures.

Fig. 2 shows the logical inference of literals $P(c,e)$ and $P(b,a)$ using complete disjunct division procedure $\Omega_1 = \langle D*_3, P(c,e), Q_1, N_1 \rangle$ and Fig. 3 shows the logical inference of literals $P(c,e)$ and $P(b,a)$ using complete disjunct division procedure $\Omega_2 = \langle D*_3, P(b,a), Q_2, N_2 \rangle$.

$\omega_1 = \langle D*_3, P(c,e), Q_1, N_1 \rangle$		
ω_1	$\neg P(u,w)$	$P(v,w)$
$P(c,e)$	1	Δ_{12}

Considering $\lambda_{12} = \{c/v, e/w\}$, we get: $\Delta_{12} = \neg P(u,e) = b_1; q_1 = g; n_1 = \{<b_1, d_1>\}$

Compare Δ_{12} with the negations of the facts: $d_1^+ = \neg P(c,b) \vee \neg P(a,e)$

$\omega_{1,1} = \langle b_1, d_1^+, q_{1,1}, n_{1,1} \rangle$		
$\omega_{1,1}$	$\neg P(u,e)$	
$\neg P(c,b)$	1	
$\neg P(a,e)$	0	

Considering $\lambda_{12} = \{a/u\}$: $\Delta_{21} = b_{1,1} = 0; q_{1,1} = 0; n_{1,1} = \{<0, I>\}$ and $d_{1,1} = 1$

Then: $Q_1 = 0; N_1 = \{E_1\}, E_1 = 0$

Fig.2. Procedures of complete disjunct division: $\Omega_1 = \langle D*_3, P(c,e), Q_1, N_1 \rangle$

$\omega_1 = \langle D^*_3, P(b,a), q_1, n_1 \rangle$		
ω_1	$\neg P(u,w)$	$P(v,w)$
$P(b,a)$	1	Δ_{12}

Considering $\lambda_{12} = \{b/v, a/w\}$, we get: $\Delta_{12} = \neg P(u,a) = b_I; q_I = g; n_I = \{b_I, d_I\}$
 Compare Δ_{12} with the negations of the facts: $d_I^+ = \neg P(c,b) \vee \neg P(a,e)$

$\omega_{I,I} = \langle b_I, d_I^+, q_{I,I}, n_{I,I} \rangle$		
$\omega_{I,1}$	$\neg P(u,a)$	
$\neg P(c,b)$	1	
$\neg P(a,e)$		1

$b_{I,1} = 1; q_{I,1} = 1; n_{I,I} = \{<1,0>\}$ and $d_{I,I} = 0$
 Then: $Q_2 = 1; N_2 = \{1\}, E_2 = 1$

Fig.3. Procedures of complete disjunct division: $\Omega_2 = \langle D^*_3, P(b,a), Q_2, N_2 \rangle$

The procedure of complete division of disjuncts $\Omega_1 = \langle D^*_3, P(c,e), Q_1, N_1 \rangle$ terminates successfully with a remainder "0" indicating that the literal $P(c,e)$ is logically deducible from the special knowledge base premises. Thus the literal $P(c,e)$ is unacceptable and is excluded from the disjunct of the additional premise, which takes the following form: $D'_6 = P(b,a)$.

The procedure of complete division of disjuncts $\Omega_2 = \langle D^*_3, P(b,a), Q_2, N_2 \rangle$ fails (with formation of "singular" remainders), so the literal $P(b,a)$ is not a logical consequence of the initial knowledge base. Thus, the literal $P(b,a)$ is considered acceptable for specifying it as a complementary premise: $D'_6 = P(b,a)$. In other words, if it is added to the initial premise base, then the deductive conclusion $d_1 = O(e,c)$ will be successful.

4. Conclusion

Methods of deductive and abductive reasoning are indispensable for designing high-performance knowledge processing systems in general and complex intelligent information-management systems in particular. Such mechanisms, in contrast to the widespread neural network approach, allow the module of explanation of the process of formation of the resulting "reasoning" by the

artificial intelligence system to be implemented. Besides, abductive methods form necessary additional premises, extending the knowledge base, developed by expert.

The abductive logical inference considered in the article, in addition to explaining the course of the logical inference with the help of inference schemes, allows you to check the correctness of interpretation of literals of additional premises. At the same time the mechanism of inability to add "illogical" statements to the base is implemented, taking into account the existing ontology of a particular domain.

A special knowledge base for checking the correctness of interpretation of literal additional premises generated by abductive logical inference is formed in the process of additional analysis of the domain and its formal description by a knowledge expert. In particular, for problems similar to the one considered in the example, the following premises can be included in the auxiliary knowledge base: $D_8 = P(x,x)$ ("self-direct"); $D_9 = O(y,y)$ (self-report) and others.

It is reasonable to limit the deductive logical inference of additional premises literals from the special knowledge base by depth by specifying the maximum permissible number of inference steps. At the same time, it should be taken into account that in this case the determination of the literal (additional premises) conclusion impossibility from a special knowledge base is determined only at an inference depth not exceeding the specified one.

Abductive logical inference by division of disjuncts allows one to form additional premises that are not only separate literals (facts), but also multiliterals (private rules). Private rules consist of literals where variables are replaced by constants. Rules generated by abductive logical inference and consisting of several literals may describe certain relations between subjects. There may be situations where each of the literals of an additional premise has a correct interpretation but the rule it defines is unacceptable for the problem being solved. In this case it may be expert necessary to check the correctness of the rules generated by abductive logical inference in a given domain.

References

1. Thacker J.: The Age of AI: Artificial Intelligence and the Future of Humanity. Zondervan (2020)
2. Chen, Y., Duong, T.Q.: Industrial networks and intelligent systems. In: Proceeding of 3rd International Conference, INISCOM 2017, Ho Chi Minh City (2017)
3. Al-Emran, M., Shaalan, H., Hassanien, A.: Recent advances in intelligent systems and smart applications. SSDC, vol. 295, pp. 46–54. Springer, Cham (2020)
4. Szczerbicki, E., Sanin, C.: Knowledge management and engineering with decisional DNA. Springer International Publishing, Switzerland (2020)
5. Putzky, P., Welling, M.: Recurrent inference machines for solving inverse problems: Under Review as a Conference Paper at ICLR 2017. University of Amsterdam (2017)
6. Czarnowski, I., Howlett, R., Jain, L.: Intelligent decision technologies 2019. Springer

Singapore (2019)

7. Khemani, D.: Artificial intelligence: knowledge representation and reasoning. IIT Madras, https://swayam.gov.in/nd1_noc20_cs30/preview, last accessed 2022/10/16
8. Rahman, S.A., Haron, H., Nordin, S., Bakar, A.A., Rahmad, F., Amin, Z.M., Seman, M.R.: The decision processes of deductive inference. *Adv. Sci. Lett.* 23(1), 532–536 (2017)
9. Meltssov, V., Zhukova, N., Strabykin, D.: Logical inference in predicate calculus with the definition of previous statements. In: Sharma, H., Saraswat, M., Yadav, A., Kim, J.H., Bansal, J.C. (eds) Congress on Intelligent Systems. CIS 2020. Advances in Intelligent Systems and Computing, vol 1334. Springer, Singapore (2021)
10. Vagin, V., Derevyanko, A., Kuteпов, V.: Parallel-inference algorithms and research of their efficiency on computer systems. *Sci. Tech. Inf. Process.* 45(5), 368–373 (2018)
11. Strabykin, D.A.: Logical method for predicting situation development based on abductive inference. *J. Comput. Syst. Sci. Int.* 52(5), 759–763 (2013)
12. Schaeken W., De Vooght G., Vandierendonck A., D'Ydewalle G.: Deductive Reasoning and Strategies. Taylor & Francis eBooks, New Jersey (2014)
13. Meltssov, V., Kuvaev, A., Zhukova, N.: Knowledge processing method with calculated functors. In: Arseniev, D., Overmeyer, L., Kälviäinen, H., Katalinić, B. (eds.) Cyber-Physical Systems and Control. CPS&C 2019. LNNS, vol. 95. Springer, Cham (2020)
14. Dolgenkova, M.L., Chistyakov, G.A.: Sequences inference method out of new facts by disjuncts division in predicate calculus (Example). <https://zenodo.org/record/57859>
15. Czarnowski, I., Howlett, R., Jain, L.: Intelligent Decision Technologies 2019. Springer Singapore (2019). <https://doi.org/10.1007/978-981-13-8303-8>
16. Rahman, S.A., Haron, H., Nordin, S., Bakar, A.A., Rahmad, F., Amin, Z.M., Seman, M.R.: The decision processes of deductive inference. *Adv. Sci. Lett.* 23(1), 532–536 (2017)

An Investigation and Observational Remarks on Conventional Sign Language Recognition

Thouseef Ulla Khan¹ and Dileep M R²

¹Department of Master of Computer Applications,
Vidya Vikas Institute of Engineering and Technology,
Mysuru, Karnataka, India,
VTU – Research Centre,
Department of Master of Computer Applications,
Nitte Meenakshi Institute of Technology,
Yelahanka, Bengaluru, Karnataka, India
1thouseef588@gmail.com

²Department of Master of Computer Applications,
Nitte Meenakshi Institute of Technology,
Yelahanka, Bengaluru, Karnataka, India
2dileep.kurunimakki@gmail.com

Abstract. In today's world, Communication of speech and hearing impaired person has been facilitated with different technologies. In the world, there are about 300 million people are deaf, 285 million are blind and 1 million are dumb, as per the World Health Organization.

Sign Language is a tool to communicate with speech and hearing impaired people. Recognition of Sign Language is a challenge for researchers from many years that have to be implemented as a system for various sign languages. Each system has its own limitations and difficult to be used commercially. Researchers have done their research in various ways to simplify the recognizing of sign languages with limited database. Researchers are trying to do research with their own large database. Through this paper, we review the different sign language recognition approaches and try to find the best method that has been used. This helps the researchers to retrieve more information to develop the Sign Language Recognition systems using current and advanced technologies in future.

Keywords: Sign language, image processing, convolutional neural network, support vector machine, deep learning, machine learning.

1 Introduction

The system consists of image processing, sign languages and sign language recognition steps. Firstly, image processing consists of image acquisition, enhancement, restoration, segmentation and object recognition stages. Sign languages are the way of communication among speech and hear impaired persons, which are represented in different languages in different geographical areas. Sign language recognition is a process of recognizing hand gestures through

the systems developed using different technologies. The following subsection 1.1 through 1.3 explains the stages involved in the process of sign languages recognition.

1.1 Image Processing

Manipulation of digital images using computer is called digital image processing. It concentrates on constructing computer system which is capable of image processing. Digital image is an input of the system, it processes image as an input using different algorithms, and produces an image as an output. Images are processed using different formats like gray scale, RGB and RGBA. Digital Image processing involves the following steps image acquisition defines the position & intensity quantization with conversion to binary digits; image enhancement defines manipulation of image pixel or wavelet transform; image restoration defines removing sensor noise; image segmentation defines partition an image into its constituent parts; representation defines the final output as a raw pixel data.

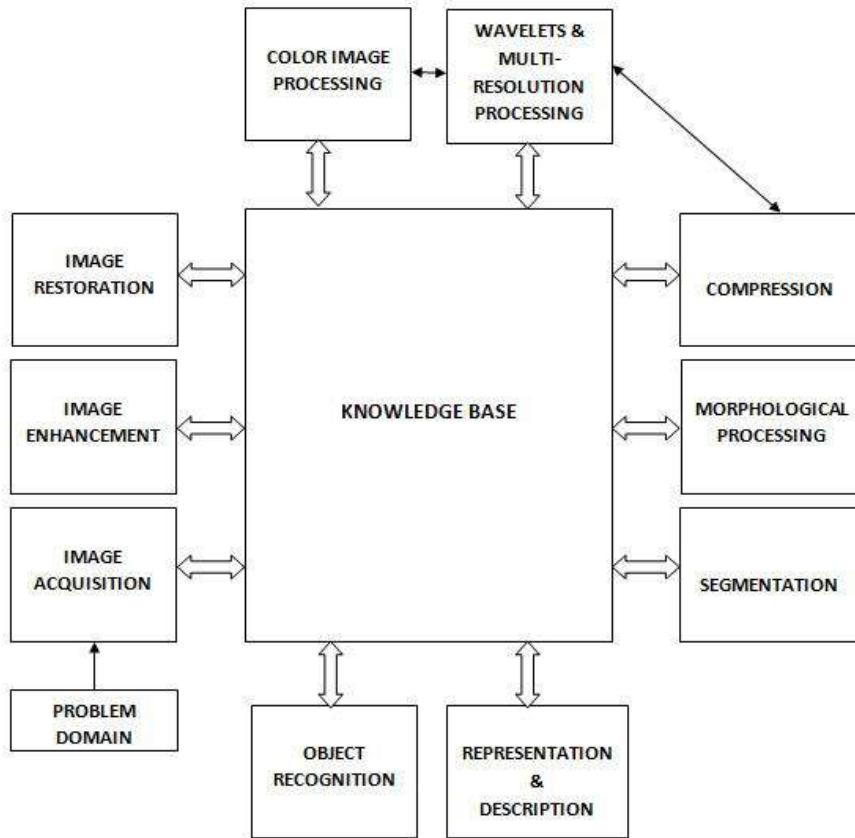


Fig. 1. Steps in Digital Image Processing

1.2 Sign Language

Around the world, more than 300 sign languages are there. They differ from nation to nation. The same language is spoken in the countries, but the sign language can have various regional accents which bring variations to use and understanding of signs among the people. Presently, sign language recognized as a formal language in 41 countries of the world. International Day of Sign Languages has been celebrated on 23rd September. Sign languages are available for alphabets, numbers, words and sentences. It can be differentiated by one hand or two hands gestures, for instance, ASL is one hand gesture where as ISL is two hands gestures. Universal sign language is not available, because it differs based on countries or regions such as America, India, Britain, France, Arab, Spain, China, Australia, Tanzania etc. Human feelings, thoughts and expressions can be

expressed by using Sign Language, also to fortify the data obtained in everyday discussions through gestures and facial expressions.

Communication is the foundation for both personal and professional relationships of all human beings. It is essential for survival in a society. For the successful verbal communication, we required a well defined language which can be understandable by all the parties. In India, Sign language is used for communication of around 26% of the disabled population. There are 6.3% of hearing impaired people in India, WHO. They use hand gestures for communication. Unfortunately, the semantics of these hand gestures are not known to majority of Indians.

Sign Language Recognition (SLR) System acts as a bridge to fill the gap of communication among the normal and the dumb and deaf people. Efficient sign language recognition (SLR) system helps to communicate among normal people and speech and hearing impaired people. The majority of the normal people do not recognize sign language, so this particular dumb and deaf population is overlooked in the world.



Fig. 2. Signs of American Sign Language Alphabets

1.3 Sign Language Recognition

The different techniques are used to recognize the sign languages like Convolutional Neural Network, Deep Neural Network, Decision tree, Artificial Neural Network, Arduino Circuit Boards, K-Nearest Neighbour, Raspberry Pi, Support Vector Machine. The different technologies are used to recognize the sign languages like Machine learning, Python, Django, Artificial intelligence, Deep learning, Discrete Wavelet Transform (DWT), Keras, Discrete Cosine Transform (DCT), Random Forest (RF), Histograms of Oriented Gradient (HOG).

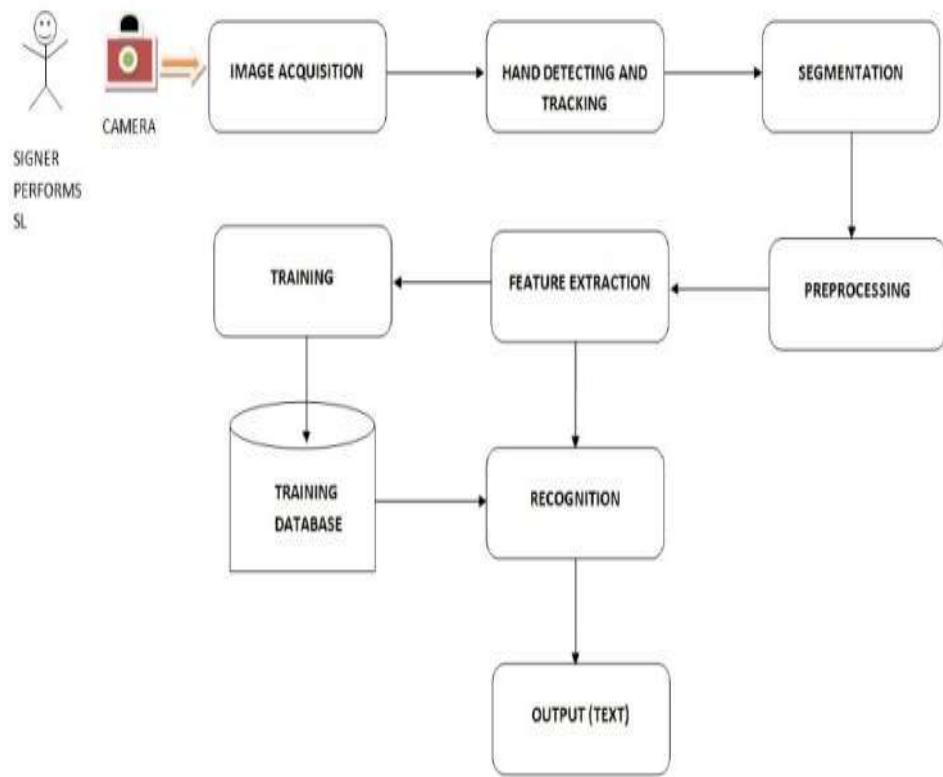


Fig. 3. Sign Language Recognition Model

2 Literature Survey

2.1 Image Processing Techniques

Collection of data is the basic step of the system, where sensors and cameras are used by many researchers to capture the hand gestures. Ajay et al. [2], used Sensor Gloves pair to detect hand gestures. Fakhar et al. [6], achieved the communication objective by using dynamic and static gestures which are identified in video based recognition system and converted into text. There are redundant Key frames present in videos, which have to be reduced by additional processing. It is a challenging task to select specific Key frames without missing needed information.

Microsoft Kinect Sensor, flex sensors, touch sensors, Nano microcontrollers, Inertial Measurement Units (IMU) and Bluetooth. Mohammad and Mehreen [13], used Web Camera to capture hand gesture images and the system predicts and displays the image name. Radha et al. [16], used Web camera to capture the hand gesture.

Sharma S et al. [18], used tri-axis gyroscopes, multichannel surface electromyogram and tri-axis accelerometers placed on forearms of the signers to capture the gestures. The Sensor Glove will convert the gestures. Sharad et al. [21], developed the SHAROJAN BRIDGE that will make use of the Wearable Technology.

Two types of feature extraction methods are available from the images captured by Web Camera in recognition of sign, that are Contour based and Region based shape representation and description methods. Skin color modelling technique, whose range is predetermined from non-pixels (background) and extract pixels (hand).

Lean at al. [10], demonstrated about how sample datasets are captured using vision based technique as shown in the Figure 4.

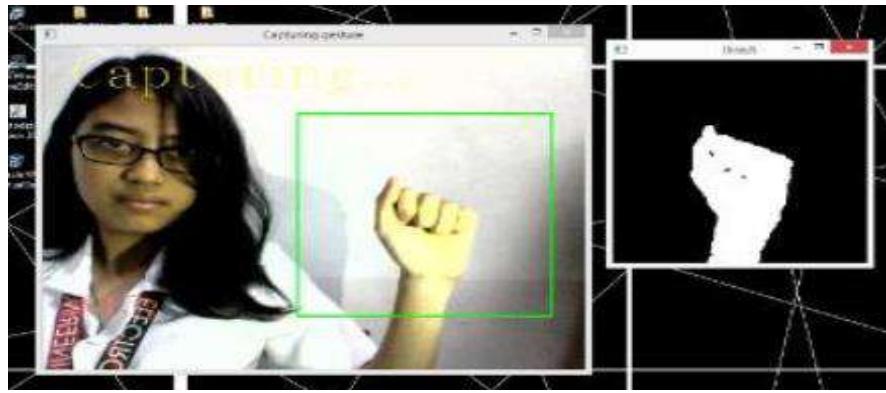


Fig. 4 . Hand Signs are captured using web camera

Mahesh [11], demonstrated about image preprocessing by image segmentation and morphological filtering as shown in the Figure 5.

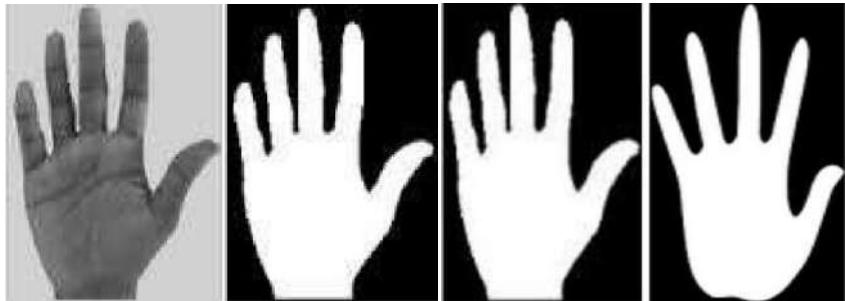


Fig. 5. Image Segmentation and Morphological filtered Image

2.2 Sign language Recognition Techniques

Abdelbasset et al. [1], demonstrated how deep learning architecture to correlate sign language video sequences, which includes two methods 2DCRNN and 3DCNN. To build the relationship between frames, they isolate features with a recurring network pattern using 2DCRNN. Ajay et al. [2], used Sensor gloves to capture the motion and values are passed to machine learning algorithm for classification to find the gestures. Ashish et al. [3], investigated the three models: a pre-trained VGG-16 including fine tuning, a VGG-16 including Transfer

learning and Hierarchical Neural Network are examined based on trainable vectors count. A Conversation model was built to convert concurrent gestures to speech and speech to gestures in Django.

Bineet et al. [4], used Krawtchouk moment-based shape features to understand the different shapes of hand gestures. Emely et al. [5], demonstrated the System that recognizes manual gestures, facial expressions and other markers, which uses Action Units (AU) recognition architecture that merges SqueezeNet and features based on geometry. Fakhar et al. [6], utilized DWT, DCT and HOG for unique Key frames of video extraction. SVM, KVM, Bagged tree and Boosted tree ensemble method are used for classification of hand gestures. Jayesh and Jyothi [7], employed an algorithm designed using Microsoft Kinect Sensor to identify and segment the hand region. CNN implemented to develop features from sign language automatically. Jayesh et al. [8], used Supervised Machine Learning for extracting and training the image features. ORB (Oriented Fast and Rotated Brief) is used for comparing the images with various classification techniques to gain optimum results.

Kasian and Hassan [9], explained sign language translation can be achieve by Machine Learning techniques. By using combined 5x2cv F test, the performance of CNN and SVM are compared through image recognition by converting the sign language. Lean et al. [10], demonstrated Convolutional Neural Network used to classify the images, where Keras is used to train the Images. Mahesh [11], has made detailed study on Eigen values and Eigen vectors which are extracted using Linear Discriminant Analysis (LDA) algorithm for gesture recognition. Mitali et al. [12], explained several Machine Learning algorithm techniques are used in ISLR system, where RF algorithm gives highest efficiency out of six machine learning algorithms. Mohammad and Mehreen [13], designed HSV colour algorithm to identify hand gestures with background black; CNN for training and classifying the images.

Neha et al. [14], constructed NAO humanoid robot is used to communicate with deaf person, which extracts uniform shape feature of hand gestures using Wavelet descriptor and classifying those gestures using Possibility Theory, which are combined for generating text sentences. These sentences are further converted to speech using NAO robot. Qazi and Mohammad [15], described that Deep Learning is a subset of Machine Learning used to find hand gestures in a collection of video frames. VGG-16 and RNN-LSTM model is used to recognize the sign language. Radha et al. [16], explained Linear Kernel Multiclass SVM algorithm used to differentiate the one handed and two handed alphabets gestures.

Rajyashree et al. [17], explained Raspberry Pi identifies and converts the sign languages into text and audio sounds.

Sharma S et al. [18], used Transfer Learning algorithm such as Trbaggboost and Conventional Machine Learning algorithms such as RF, SVM, decision trees as main learners. The outcomes for distinguish of signs by Trbaggboost, yields highest accuracy including the labelled data. Sharma S et al. [19], collected data using 6 degree of freedom inertial measurement units (IMUs) on both hands. Deep Transfer Learning consists of CNN, two (Bi LSTM) layers and CTC to sentence recognition.

Sakshi and Sukhwinder [20], developed a Computer-Vision based model which uses CNN for feature extraction and classification of gestures. Sharad et al. [21], designed Arduino circuit boards and Texas instrumentation circuitry that are used to recognize sign language. Raghuveera T et al. [22], demonstrated that a system uses Microsoft Kinect with RGB images to capture the gestures with finger spelling, double and single-handed signs. To extract the hand features, HOG and local binary patterns are used.

Mahesh [11], demonstrated the sign language recognition process as shown in the Figure 6.

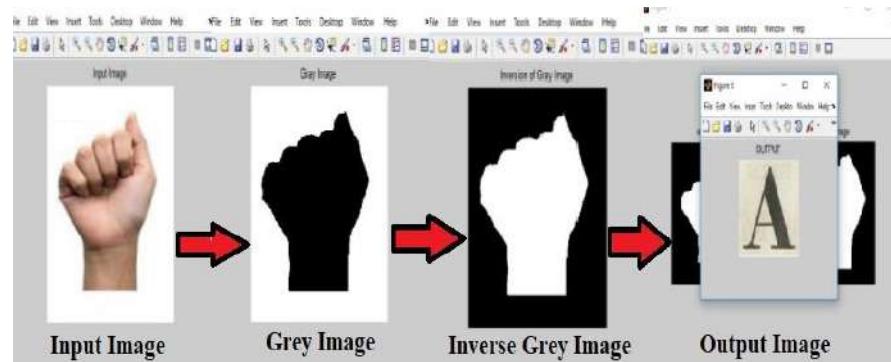


Fig. 6. Sign Language Recognition Process

From this, we can learn that many technologies are used to identify the Sign Languages and to convert them, so that it will facilitate the communication among dumb and deaf people with normal people.

3 Comparative Analysis

The research work has been done on different Sign Languages with different technologies, which results in different accuracies. The accuracy of the Sign Language Recognition Systems using CNN is 99.3%, CNN with Computer Vision is 90%, CNN with Recursive Neural Network 92%, CNN with SVM is 96%, Deep Learning 98%, Deep Neural Network is 93.67%, Trbaggboost is 97.04%, Facial action is 88%, Key-Frame Selection is 97.5%. This comparative analysis is explained in detail.

3.1 Efficiency Comparisons with Various Parameters

Abdelbasset et al. [1], concluded the results achieved with accuracy of 92% by 2DCRNN and 99% by 3DCNN through the fourfold cross-validation technique, F1 score, and AUROC. Ashish et al. [3], designed the model with 98.52% accuracy for one-hand and 97% accuracy for two-hand gestures. Emely et al. [5], applied the model which achieved an accuracy of 88% for 119 classes, with the combination of art of gesture recognition.

Fakhar et al. [6], obtained accuracy of 95.6% on 100 common words with overall accuracy of 97.5% on 37 Urdu alphabets. Jayesh and Jyothi [7], developed the system that achieves 99.3% of accuracy to recognize the gestures. Jayesh et al. [8], achieved 93.26% of recognition accuracy by ORB tuning with K-NN classification.

Kasian and Hassan [9], valued the system, in which CNN scored 96% in three parameters such as precision, accuracy and recall, while SVM got same rate in precision but poor on accuracy and recall parameters. Lean et al. [10], analyzed that system has achieved a testing accuracy of 93.67% on an average, of which alphabet recognition accuracy is 90.04%, number recognition accuracy is 93.44% and static word recognition accuracy is 97.52%.

Mitali et al. [12], established that RF Machine Learning algorithm has given the maximum accuracy of 98.44% out of six Machine Learning algorithms. Mohammad and Mehreen [13], system has achieved above 90% of accuracy. Qazi and Mohammad [15], evolved the model that achieved 98% accuracy on Dataset of Sign Language Hand Gestures. Radha et al. [16], designed a model with 95% accuracy. Trained linear kernel Multiclass SVM models were used to distinguish the one handed alphabets with 56% accuracy and two handed alphabets with 60% accuracy.

Sharma S et al. [18], admitted that from the new subject when two observations of labelled data are included with existing system's training data, we get an average

accuracies of TrAdaboost is 71.07%, TrResampling is 72.92%, TrBagg is 76.10% and RF is 76.79% for classifications. If the number of labelled data is same, then Trbaggboost yields 80.44% of an average classification accuracy. If we increase the number of labelled data, then Trbaggboost yields up to 97.04% of classification accuracy.

Sharma S et al. [19], demonstrated that when the number of observations reduced from 10 to just 3 of each sentence available for training the model, where noted degradation in an average is 54% of accuracies obtained without Transfer Learning. But, with Deep Transfer Learning approach the degradation will be reduced up to 11.5%. Sakshi and Sukhwinder [20], achieved accuracies of 92.43%, 88.01%, and 99.52% by using three sets of data respectively. The terms like recall, precision, time consumption and f-score are used to evaluate the efficiency of the method. The proposed method has achieved good results comparing to the existing method. Raghuveera T et al. [22], achieved an accuracy of 100% on 9, A, F, G, H, N, P signs and 71.85% as an average accuracy improved by the SVM trained three feature classifiers.

Lean et al. [10], discussed about calculation of accuracy by using the formula

$$\text{Accuracy rate} = \frac{\text{Total nos. of correct recognized letters from users}}{(\text{Total no. of users})(\text{No. of Trials})}$$

They calculated the accuracy rate of each letter by using the above formula, where the total number of the correctly recognized letters from all the users from all the trials was totaled and divided by the total number of samples, which is the number of users (30) multiplied by the number of trials as shown below Table 1.

Table 1. Letter Recognition Accuracy

Letter	Correctly Recognized Gestures	Incorrect Recognized Gestures	Accuracy (%)	Ave. Time (s)
A	90	0	100	2.02
B	82	8	91.11	4.01
C	90	0	100	2.2
D	90	0	100	2.46
E	87	3	96.67	3.59
F	84	6	93.33	4.95
G	89	1	98.89	2.85
H	76	14	84.44	5.39
I	85	5	94.44	3.37
J	81	9	90	4.57
K	85	5	94.44	4.06
L	89	1	98.89	2.38
M	74	16	82.22	5.98
N	81	9	90	3.92
O	81	9	90	3.62
P	78	12	86.67	4.97
Q	86	4	95.56	3.76
R	71	19	78.89	6.24
S	78	12	86.67	4.6
T	67	23	74.44	7.62
U	77	13	85.56	5.11
V	76	14	84.44	5.07
W	87	3	96.67	2.46
X	78	12	86.67	4.98
Y	84	6	93.33	3.5
Z	81	9	67.78	8.31
Overall Rating			90.04%	4.31

4 Observations and Discussions

The research to develop the system for Sign Language Recognition has been limited to alphabets, numbers and words. It has been translated from sign language to text or vice versa. Each system has been observed and discussed by comparing the accuracies developed based on different methodologies as shown in the Table 2. The observation extended to different sign languages with accuracies as shown in Table 3. From this we can find the efficiencies of the systems.

Table 2. Comparison of accuracies based on different methodologies

Author	Method	Accuracy	Publication Year
1	CNN , Recursive Neural Network	92%	2021
3	Deep Neural Network	98.52%	2020
5	Facial Action	88%	2021
6	Key Frame Selection	97.5%	2020
7	CNN	99.3%	2020

8	Kinect Sensor, ORB, KNN	93.26%	2020
9	SVM , CNN	96%	2021
10	Deep Learning- CNN & Keras	93.67%	2019
12	Machine Learning	98.44%	2021
13	3 layers CNN & Computer Vision	90%	2020
15	Deep Learning	98%	2021
16	Machine Learning- Multiclass	95%	2020
18	Trbagboost	97.04%	2020
19	Deep Transfer Learning	54%	2021
20	Deep Learning	92.43%	2021
22	Microsoft Kinect	71.85%	2020

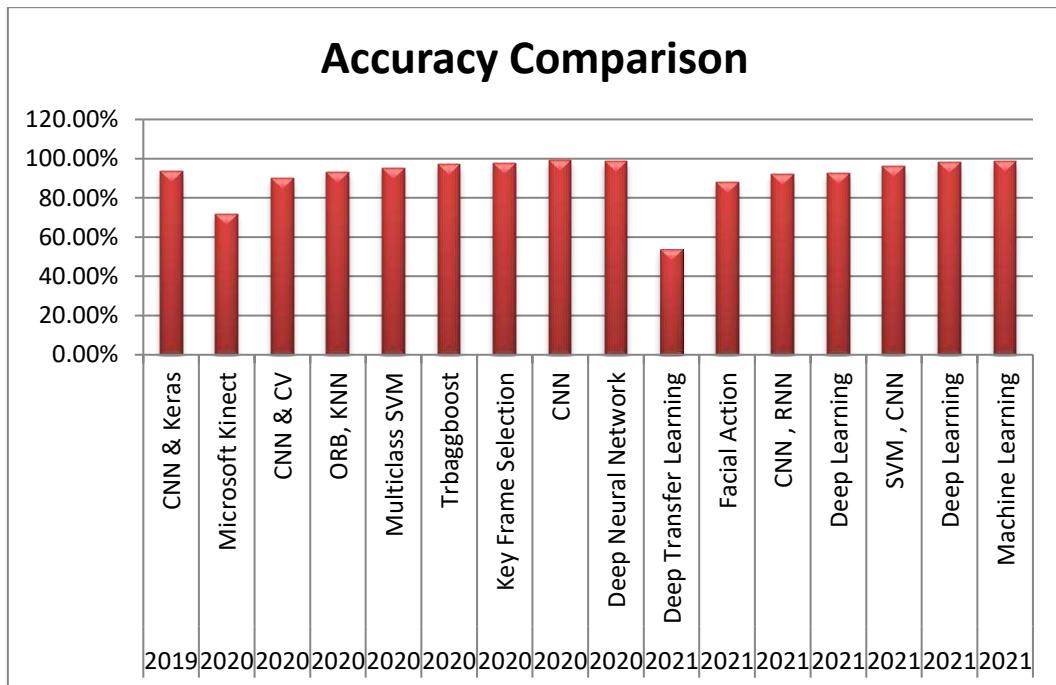


Fig. 7. A graph representation of accuracies comparison

4.1 Comparison of an accuracy based on different Sign Languages

Table 3. Comparison of accuracies based on different sign languages

Sign Language	Accuracy
---------------	----------

Indian	98.52%
American	93.67%
Tanzanian	96%
Arabic	92%
Pakistan	97.5%
Brazilian	88%

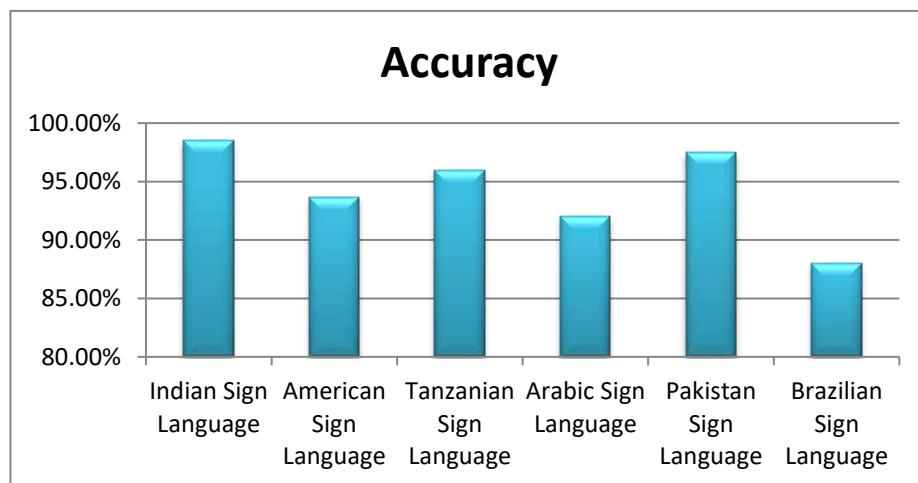


Fig. 8. Graph of accuracies of different sign languages

5 Proposed Method

Deep feed forward network concept called Multilayer Perceptrons (MLPs) will be used for sign recognition. The flowchart of a proposed method to recognize sign language as shown in the Figure 9.

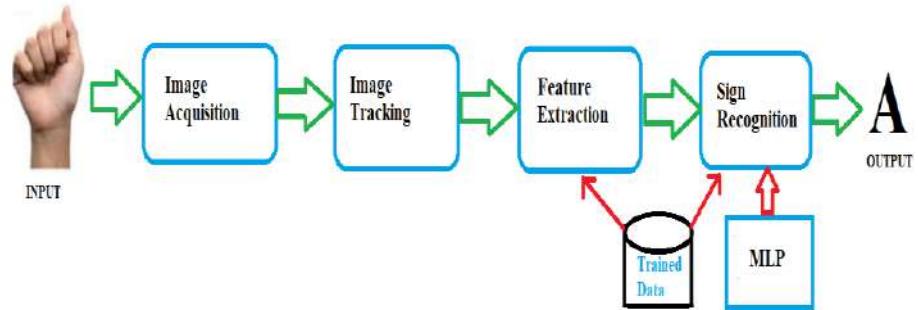


Fig. 9. Block diagram of proposed system

Image Acquisition:

This is the first step of the process. It can be achieved by using web camera.

Image Tracking:

Image preprocessing will be performed to improvise the image for the further process like Image enhancement, filtering, smoothing, sharpening and compression.

Feature Extraction:

There are mainly two types of feature extraction methods are Contour based shape recognition and Region based shape recognition popularly used in sign recognition. Image segmentation, morphology, object representation and description are used to extract features of an image.

Sign Recognition:

Multilayer Perceptrons is a Deep Feedforward Neural Network with multiple layers of perceptrons that have activation functions such as , ReLus, Sigmoid functions, and tanh. It consists of Input layers, Output Layers and Hidden Layers which help to recognize the images as shown in the Figure 10.

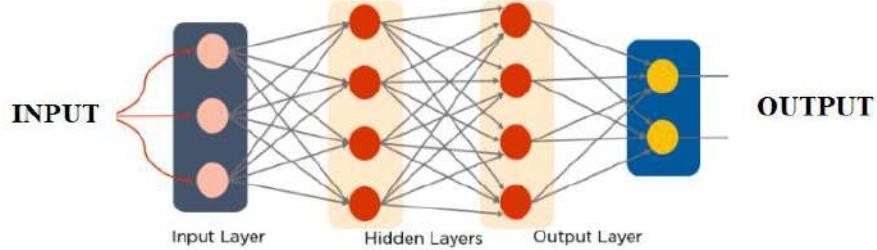


Fig. 10. Multilayer Perceptrons

MSE loss function is evaluated on the whole training set by using the below formula, for the given function $f(x, \Theta)$ with the target function $f^*(x)$

$$J(\boldsymbol{\theta}) = \frac{1}{4} \sum_{x \in \mathbb{X}} (f^*(x) - f(x; \boldsymbol{\theta}))^2 .$$

The proposed system would perform with 91% of accuracy with limitations of datasets on different sign languages.

6 Conclusion

Several efforts have been made to develop the System for Sign Language Recognition with improving efficiencies by using different technologies on different Sign Languages.

Only alphabets and static signs were classified by the sign language recognition system. It has been motivating towards the system recognizing the dynamic gestures. Additionally researchers are finding the large database of different sign language signs. Researchers are working on their own database with small vocabulary. There is a deficiency of large database of sign languages in some of the countries. The classification methods for recognize the signs differs from one researcher to another. It is still subjective for comparing one method to another for the researchers by using their own innovative ideas and limitations to identify the sign languages. In most of the counties, there is a variation of sign languages due to variation in gestures and grammar. So it is difficult to the researcher to have straightforward comparison between different approaches using different technologies. But still we have to do investigation for developing more efficient systems with emerging technologies. This system can be used in the schools, colleges, offices and public places. It facilitates the communication between dumb

and deaf person and normal person more efficiently. It increases teaching/learning capability of speech and hearing impaired person, so that they can easily communicate with the society and live a better life in future.

7 Future Scope

By using current advanced technologies, we can improve the efficiencies of current systems and explore to different sign languages. It can be extended to conversion of Sign language to Oral Language or vice versa, to improve the communication between general person and speech & hearing impaired person. The Sign Language Recognition System that can translate sign language to text, sign language to audio, audio to sign language, text to sign language. This system helps to communicate with speech and hearing impaired person in public places like airport, railway station, hospital, police station, education, court, government office; and to handle the devices like smart television, smart phone, computer, laptop, tablet by implementing advanced technologies on different sign languages. The mobile applications, desktop applications and web applications could be developed for easy communication of speech and hearing impaired person in public places.

References

1. Abdelbasset Boukdir, Mohamed Benaddy, Ayoub Ellahyani, Othmane ElMesloui and Mustapha Kardouchi : Isolated Video-Based Arabic Sign Language Recognition Using Convolutional and Recursive Neural Networks. (2021)
2. Ajay S, Ajith Potluri, Sara Mohan George, Gaurav R, Anusri S : Indian Sign Language Recognition Using Random Forest Classifier. (2021)
3. Ashish Sharma, Nikita Sharma, Yatharth Saxena, Anuraj Singh and Debanjan Sadhya : Benchmarking deep neural network approaches for Indian Sign Language recognition. (2020)
4. Bineet Kaur, Garima Joshi and RenuVig : Indian sign language recognition using Krawtchouk moment-based local features. (2017)

- 5.** Emely Pujólli da Silva, Paula Dornhofer Paro Costa, Kate Mamhy Oliveira Kumada and José Mario De Martino : Facial action unit detection methodology with application in Brazilian sign language recognition. (2021)
- 6.** Fakhar Ullah Mangla, Aysha Bashir, Ikram Lali, Ahmad Chan Bukhari and Basit Shahzad : A novel key-frame selection-based sign language recognition framework for the video data. (2020)
- 7.** Jayesh Gangradeand Jyoti Bharti : Vision-based Hand Gesture Recognition for Indian Sign Language Using Convolution Neural Network. (2020)
- 8.** Jayesh Gangrade, Jyoti Bharti and Anchit Mulye : Recognition of Indian Sign Language Using ORB with Bag of Visual Words by Kinect Sensor. (2020)
- 9.** Kasian Myagilaand Hassan Kilavo : A Comparative Study on Performance of SVM and CNN in Tanzania Sign Language Translation Using Image Recognition. (2021)
- 10.** Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac, Maria Abigail B. Pamahoy, Joni Rose R. Forteza, and Xavier Jet O. Garcia : Static Sign Language Recognition Using Deep Learning. (2019)
- 11.** Mahesh Kumar N B : Conversion of Sign Language into Text. (2018)
- 12.** Mitali Potnis, Divya Raul, Madhura Inamdar : Recognition of Indian Sign Language using Machine Learning Algorithms. (2021)
- 13.** Mohammad Elham Walizad and Mehreen Hurroo : Sign Language Recognition System using Convolutional Neural Network and Computer Vision. (2020)
- 14.** Neha Baranwal, Avinash Kumar Singh and Nandi G. C : Development of a Framework for Human–Robot interactions with Indian Sign Language Using Possibility Theory. (2017)
- 15.** Qazi Mohammad Areeb and Mohammad Nadeem : Deep Learning Based Hand Gesture Recognition for Emergency Situation: A Study on Indian Sign Language. (2021)
- 16.** Radha S. Shirbhate, Vedant D. Shinde, Sanam A. Metkari, Pooja U. Borkar, Mayuri A. Khandge : Sign language Recognition Using Machine Learning Algorithm. (2020)

- 17.**Rajyashree, Deepak O, Naresh Rengaswamy, Vishal K S : Communication Assistant for Deaf, Dumb and Blind. (2019)
- 18.**Sharma S, Gupta R and Kumar A : Trbaggboost: an ensemble-based transfer learning method applied to Indian Sign Language recognition. (2020)
- 19.**Sharma S, Gupta R and Kumar A: Continuous sign language recognition using isolated signs data and deep transfer learning. (2021)
- 20.**Sakshi Sharma and Sukhwinder Singh : Recognition of Indian Sign Language (ISL) Using Deep Learning Model. (2021)
- 21.**Sharad Agarwal, Falgun Patel, Prakhar Chaturvedi, Dr.Asha S : A Novel Approach for Communication among Blind, Deaf and Dumb People. (2018)
- 22.**Raghuveera T, Deepthi R, Mangalashri R and Akshaya R: A depth-based Indian Sign Language recognition using Microsoft Kinect. (2020)

Exploratory Project of Digital Twin Technology in Cyber-physical Systems

Irony Nunes de Oliveira, Hanameel Carlos Vieira Gomes, Kelipys da Silva Firmino, Antonio Eduardo Carrilho da Cunha, Cícero Roberto Garcez
Laboratório de Segurança Cibernética de Sistemas Ciberfísicos,
Instituto Militar de Engenharia, Praça General Tibúrcio n° 80, Rio de Janeiro (RJ), Brazil
nunes.irony@ime.eb.br
gomes.hanameel@ime.eb.br
kelipys.silva@ime.eb.br
carrilho@ime.eb.br
garcez@ime.eb.br

Abstract. Within the timeline of technology evolution, we have reached the age of Industry 4.0. This era encompasses advanced technology systems such as artificial intelligence, robotics, the internet of things, cloud computing, cyber-physical systems, and digital twins. Digital twins are used as real-time emulations of tangible assets, such as cyber-physical systems, for simulations, analysis, and decision-making, creating industrial optimization scenarios and even malicious attacks on major global institutions. This work aims, through an experimental project using system engineering, to focus on the analysis of how the Cyber-Security of Cyber-Physical Systems Laboratory (LaSC), located within the Instituto Militar de Engenharia (IME), relates to other institutions that make use of its high technological capacity regarding the application of digital twins in cyber-physical systems.

Keywords: Digital twin, cyber-physical system, industry 4.0, cybernetics.

1 Introduction

The First Industrial Revolution began at the end of the 18th century. It has the characteristic of bringing technologies such as, for example, the steam engine, the spinneret, the spinning machine, and the process of refining iron to be used in metallurgy developed by Henry cut. In summary, this period replaced manual tools with machines.

The Second Industrial Revolution began around a hundred years later. The development of electricity marked it, the expansion of the telegraph, the manufacture of chemical products guided by science, the emergence of the combustion engine, and the invention of the automobile and the plane, technologies that permeated industrialization worldwide. [1]

The Third Industrial Revolution appeared in the 20th century. As strengths advance in the field of electronics, in the development of computers for personal use, robots in production lines, and the emergence of the world wide web, the internet. At this point in history, the works people once did now have robots. [1]

The beginning of the Fourth Industrial Revolution comes from the introduction of IoT (Internet of Things) in manufacturing environments. Companies worldwide are building their machines, storage systems, and production facilities with Cyber-Physical Systems (CPS). Making an intelligent system appear in which consumers, factories, and devices form a single dynamic and organized network based on information available on the web brings the philosophy of innovation into the industry.

Cyber-physical systems are computational and collaborative systems whose operations are monitored, coordinated, controlled, and integrated by communication and computing cores. Just as the internet has transformed the way humans interact with each other, cyber-physical systems will change how we interact with the physical world around us. Many significant challenges are in economically vital domains such as transportation, healthcare, manufacturing, agriculture, livestock, energy, defense, construction, and others.

Nowadays, the best decisions are made based on digitizing and treating large volumes of data. One of the technologies that help in making the best decisions is the Digital Twin. In general, a digital twin is a model that emulates in real-time the behavior of a given physical entity for observations, simulations, analyses, and decision-making. [2]

With this context, this work brings, with a broader view, an exploratory project about the use of digital twins in cyber-physical systems. The study focuses on using such technologies by the Cybernetic Security Laboratory of Cyber-physical Systems (LaSC), located at the Instituto Militar de Engenharia (IME).

2 Conceptual Foundation

This chapter presents the main definitions and concepts involved with digital twin technology and cyber-physical systems. From the understanding that the exploratory approach of digital twin technology in cyber-physical systems is a system of systems, it is initially necessary to consider systems engineering.

2.1 System Engineering

Systems Engineering consists of an interdisciplinary approach to mastering the technical and managerial effort required to transform a set of stakeholder needs and expectations, and constraints from a problem into a solution and the support of this solution during its life cycle, which comprises from design to disposal [3]. It is not the only definition of Systems Engineering, but it agrees with the definition given by the International Council on Systems Engineering (INCOSE), which: System Engineering is a multidisciplinary and integrative approach that enables the design, use, and closure of successful complex systems, using system principles and concepts and management, scientific and technological methods. [4]

INCOSE is a non-profit organization created to develop and disseminate transdisciplinary principles and practices that allow successful systems operation. Connecting systems engineering professionals with educational, networking, and career advancement opportunities aimed at developing the global community of systems engineers and systems approaches to problems based on the conception of the work. [4]

The relationship between systems, the need for models and digital representations of physical products, the dawn of industry 4.0 and society 5.0, as well as the growth of cybernetics and intelligent systems [4] all this set gives rise in engineering to the need for tools capable of better responding to such demands. Systems Engineering is, in this context, adding all these capacities.

Through the Unified Modeling Language (UML), software engineering provides a graphical organization of several essential points for understanding a system through the elaboration of standardized graphic diagrams capable of demonstrating important characteristics of numerous system attributes. Originating from the UML came the Systems Modeling Language (SysML). SysML is a language whose graphical representation is essential in translating Systems Engineering based on Model-Based Systems Engineering (MBSE) models. SysML uses diagrams and tables to express system information through a standard set of nine types of diagrams, as shown in **Fig. 1**, which can organize information in a complex system.

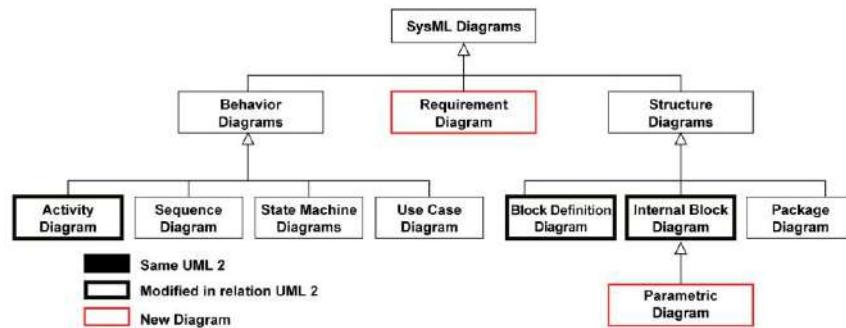


Fig. 1. SysML Metadiagram (Source: [5])

Given the above, systems engineering provides this research with tools capable of assisting in the work's preparation, design, and presentation. The following chapters and sections of this work will present the SysML diagrams used to understand and explore digital twin technology in cyber-physical systems. SysML allows to express the requirements, structure, and behavior of the system of systems using standardized images and language. As cyber-physical systems combine more than one engineering discipline, SysML seems to be a proper means of modeling complex systems and supports an interdisciplinary engineering approach. A better organization of the research, knowledge, and management domains, as well as the presentation of diagrams related to research management, technical and scientific

knowledge domains, as well as the respective diagrams, will be developed in the continuation of the conceptual foundation and in chapter 3, which deals with the broad vision of LaSC.

2.2 Digital Twin

As every action reacts, there is at least one motivating event for evolution until to develop the solution. It is excellent to remedy the effects such as the absence of machines due to maintenance, lack of cooling, lack of raw material, interruption in the supply of electricity, and monitoring of processes, among other challenges, which sometimes make production unfeasible. The industry is no different incited by the broader adverse conditions, mainly by reducing costs in its processes.

Through technology, the industry received the willingness to make industrial processes viable and optimize their production. Although according to the company Effortech, a pioneer in developing IoT solutions, the digital transformation took place in the 1960s by NASA (National Aeronautics and Space Administration), which created a digital replica of Apollo 13.

However, after all, what is the Digital Twin? According to [6] in the book Digital Twin: Possibilities of the new Digital twin technology, the digital twin is, by definition, a virtual entity identical to an equipment or process of an industry. Another definition is from General Electric (GE), which defines the Digital Twin as the software representation of an asset, system, or physical process designed to detect, prevent, predict and optimize through analysis in real-time to deliver business value.

With the Digital Twin, a real-time simulation in the natural environment with the project variables is now possible. The application is not only restricted to industry, as at Itaipu Binacional with its virtualized environment at the Instituto Militar de Engenharia (IME), but also in the daily lives of human beings on the factory floor, such as a BMW ergonomics project for the assembly line. The creation of Ericsson around a model to identify the best locations focused on installing 5G technology. The Brazilian startup MedRoom helped health professionals with 3D technology in the separation of conjoined twins and continues to be active in the market for actual training purposes virtual, among other projects. At [7], according to the website, the multinational believes that the use of the Digital Twin creates a consistent improvement in effectiveness, minimizes failure rates, shortens development cycles, and opens up new business opportunities. In other words, it creates a lasting competitive advantage.

Fig. 2 illustrates that young electromechanical engineers designed, simulated, and manufactured their new electric car in two years, thanks to the digital twin.

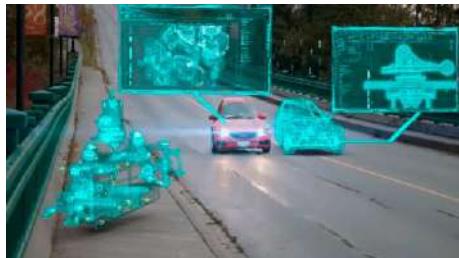


Fig. 2. Digital Twin of an electric car (Source: [7])

Fig. 3 highlights, in the construction of the Digital Twin, the understanding of the block diagram with a filled diamond that indicates that every structure will carry resources, technologies, and infrastructures, which will compose the Digital Twin.

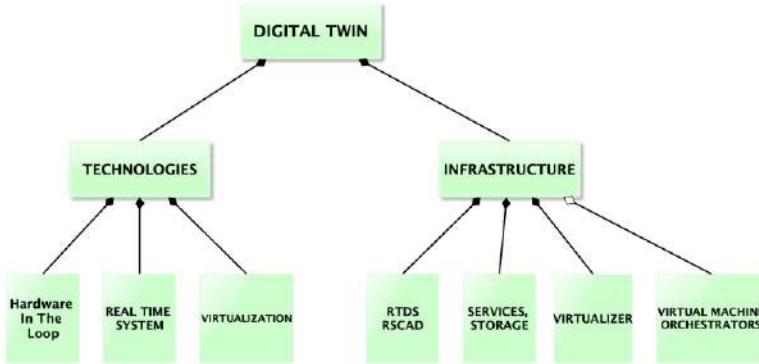


Fig. 3. Digital Twin Technology Block Diagram (Source: The Authors, 2022)

2.3 Cyber-physical Systems

To talk about cyber-physical systems, it is necessary to go back in time and involve the concepts of traditional industry and the principle of the Internet of Things (IoT). With the artisans, the manufacturer sought constant improvements in processes with the labor of workers and machines to expand its production. Consequently, profits increased, and production methods became, at the time, more complex but more efficient with the use of automation and control. Thus, starting from traditional methods, development such as motorize, mass production, digitalization, and consequently, the fourth Industrial Revolution - Industry 4.0, has been provided over the years.

This concept first appeared in 2011, cited by the German government as shown on the website of the State University of Maranhão (UEMA, 2018). In 2015 the (Reference Architectural Model for Industries - RAMI 4.0), created by engineers and presented at the Hannover World Fair in Germany, was mentioned, considered the new concept of Industry 4.0 until today.

In 1999, incredibly before the emergence of the concept of Industry 4.0, the term Internet of Things (IoT) was used by the British Kevin Ashton in his presentation to call the attention of a company about the Radio Frequency Identification Tag (RFID). [8]

The entry of IoT in industry 4.0 made possible the intelligent connection in real-time of CEOs, Directors, Managers, and Distributors, keeping them aware of the productions carried out on the factory floor without their permanence in loco. Therefore, the Internet of Things (IoT) consists of the connection between a network of physical objects, environments, vehicles, and machines through electronic devices, allowing the collection and exchange of information [9].

Still, [9] states that a cyber-physical system integrates computing and physical manufacturing processes. The relationship between communication networks, IoT sensors, actuators, computing, and storage defines Cyber-Physical Systems. This interaction provides the reality of the physical world in digital environments.

In this research, a willingness to associate with the industrial revolution 4.0, efficiency, effectiveness, and effectiveness was perceived, which establishes a massive motivation for integrating cyber-physical systems, Big Data, and Artificial Intelligence (AI), making productions highly productive and qualitative [9].

2.4 Cybernetics and Simulations

The evolution of science sees cybernetics as an evolution, where to know something is to produce a model of how the object or system works. To know is to simulate a particular form of a model that consists of reproducing the functioning of a system [10]. Nevertheless, why see precisely in cybernetics the beginnings of simulation science? Because "the faculties of the mind are always properties of information processing systems" and, as we will see, cybernetics deals with information, a general science of particular simulation systems [11].

In 1948, Norbert Wiener published a book entitled "Cybernetics: the control and communication in the animal and the machine." This book presents ideas that start from the hypothesis that the way systems

respond to messages from the environment surrounding them is equivalent and reducible to mathematical models [12].

Cybernetics attempts to understand the communication and control of machines, living beings, and social groups through analogies with electronic devices [13]. It focuses on how anything (digital, mechanical, or biological) processes information, reacts to information, and changes or can be changed to perform the first two tasks better.

Cybernetics has influenced many scientific fields, including the study of man. Several scientists were inspired to think of computer technology as an extension of human capabilities. The vast area of cybernetics includes, in addition to the study of language, forms of communication, messages between humans and between humans and machines, modeling of the man-machine prototype, nervous system, and others.

Computers helped her to represent the world in machines and significantly changed man's idea of computer technology. The development of Cybernetics led scientists to new mathematical models, increasingly complex, arising from new ways of conceiving systemic human-machine involvement.

In a more succinct definition, a communication interaction between man and machine characterizes a cybernetic event. Any natural or suspected unauthorized access to the system, such as electronic attacks or violation of privacy, including, for example, denial of service attack (DoS or DDoS); cyber terrorism; hacker attack; Trojan Horse; phishing attack; man-in-the-middle attack (MITM); application layer attack; compromised critical attack; malware (including spyware or ransomware) or computer virus infection; and cyber-attack also characterizes a cybernetic event.

Having approached cybernetics and cybernetic event, we now discuss simulation, especially real-time simulation, which is the technology used in a digital twin platform and the representation of a cyber-physical system.

Simulation is one of the most rugged system design and operation analysis tools. It allows the acceleration of the system's operation over time. It makes it possible to predict the almost inevitable accidents that occur when implementing a natural system, in addition to saving economic resources, as it eliminates the construction of prototypes for testing. Simulation can be helpful in any phase of the life cycle of a product or system: from the problem analysis and requirements definition phase to the design, implementation, and operation phases [14]. Carrying out a simulation study before implementing the existing system is very important.

Real-time simulation, applied to electrical power systems, considers some aspects that justify its use, such as:

- A controlled environment (laboratory) for realistic simulations;
- Absolute Test control, protection, or monitoring equipment used in electrical power systems; and
- Operations training for users of the electrical power system.

Thus, the simulation of electrical power systems plays a strategic role in the planning and operation of these systems and in designing the equipment that constitutes them. Network operators, equipment manufacturers, and researchers, among others, use a broad portfolio of simulation tools to carry out studies. These studies allow investigation of the impacts caused by connecting a new subsystem to the network or by adjusting the parameters of existing equipment to ensure that the system's reliability, efficiency, effectiveness, and effectiveness have not deteriorated [15].

Simulation in real-time means that the time executed in a simulation step corresponds precisely to the time of the operation simulated in real-time. That is, the real-time simulation imposes restrictions on the simulated model so that the execution time of each step of the simulation must be less than or equal to the period of the real-time operation to guarantee synchronization between the simulation and the actual simulated event [16].

Fig. 4 illustrates two situations that can arise depending on the time required for the simulation platform to complete the calculation of the state outputs for each time step:

- 1) if the execution time for the system simulation is less than or equal to the selected time step, the simulation is considered in real-time; and
- 2) if the execution time is more significant than its time step size for one or more time steps, overruns occur, and the simulation is said to be offline [16].

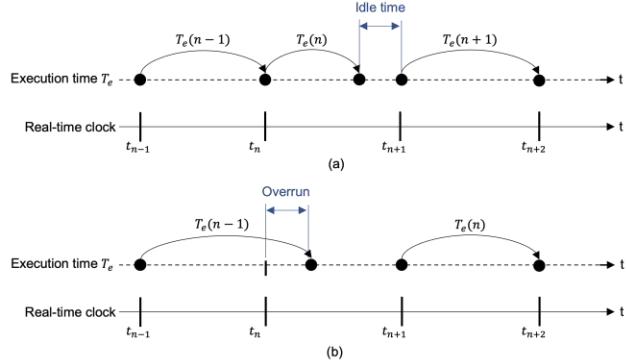


Fig. 4. Real-Time simulation and offline simulation. (a) Real-Time Simulation. (b) Offline Simulation (Source: [16])

The continuous evolution of computing technologies allowed, in the 1990s, the development of Real-Time Digital Simulators, also known as RTDS (Real-Time Digital Simulator). An RTDS of an electrical power system reproduces output waveforms (voltage or current) with the desired accuracy, representing the behavior of the simulated natural power system. To achieve this goal, an RTDS needs to solve the simulated model's equations of state in one execution time step within the same time frame as a real-world clock. In this way, the RTDS produces outputs in discrete time intervals, where it is possible to calculate the system states using a fixed time step. [16]

Therefore, RTDS is a technique for the transient simulation of power systems using a digital computer time-domain solution. It performs physical protection, control, and monitoring equipment tests in a closed loop through hardware and software. The RTDS is the hardware, and the software can be, for example, the RSCAD. It is possible to model the representative systems by taking advantage of the component models available in the RSCAD software tool library through a graphical interface and simulating on a hardware platform (RTDS) using parallel computing [16].

Real-time digital simulation applied to the power systems domain can be classified as [16]:

- 1) fully digital real-time simulation (model-in-the-loop, software-in-the-loop and processor-in-the-loop); and
- 2) real-time simulation with a simple interface (Hardware-in-the-loop).

The primary motivation for using real-time simulation, as mentioned at the beginning of the section, is to test actual equipment in a controlled environment (laboratory) for realistic simulations of simulated electrical power systems, but representative of the system that connects the hardware under test. The name of this configuration is hardware-in-the-loop (HIL) in the real world. In this exploratory project, we will only address the hardware-in-the-loop technique, as LaSC uses it.

In the simulation environment of electric power systems, the HIL configuration uses the insertion of actual equipment in the simulation information flow, where the performance of this equipment depends on the signals received from the simulation, as well as the simulation data are influenced by the real equipment sign. **Fig. 5** represents the basic schematic of a real-time HIL simulation test.

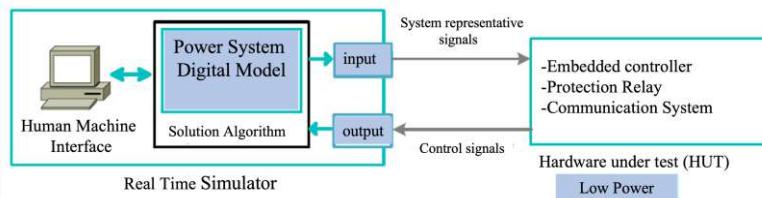


Fig. 5. The basic scheme of HIL systems (Source: [15])

There are two types of HIL simulations: controller hardware-in-the-loop (CHIL) and power hardware-in-the-loop (PHIL). In the CHIL model, the real-time model represents the entire power system, and the hardware under test (HUT) is control equipment where there is no power flow. In the PHIL configuration, the HUT is a part of the electrical power system, which exchanges power with the modeled electrical system. As the RTDS is an electronic device, the interface between the simulated model and the HUT uses power amplifiers in the PHIL configuration. **Fig. 6** illustrates the basic concepts of CHIL and PHIL. [16]

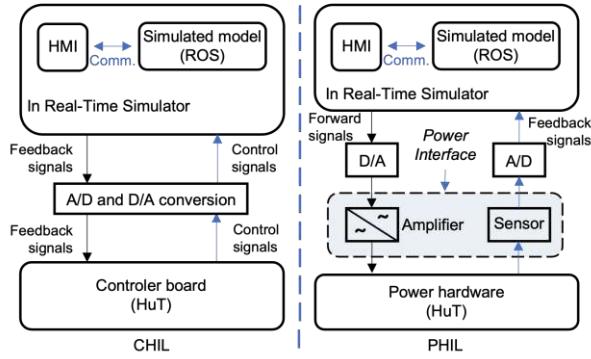


Fig. 6. CHIL and PHIL basics concepts (Source: [16])

3 An Overview of LaSC

The Cybernetic Security Laboratory of Cyber-physical Systems (LaSC) has state-of-the-art technology to add to the industrial needs of national and international companies, comprising the sectors that cover technical-operational behavior according to market demand.

According to the required project, building interactive virtual platforms in real-time is possible by linking their physical structure, thus, giving rise to the simulation model of cybernetic events and establishing the Digital Twin concept. Now available for testing, mainly with internal and external malware attacks. There is also the possibility of changing the communication networks, databases, operating systems, communication ports, and mechanisms by inserting or removing them according to the demand for the feasibility study, regardless of academic or professional nature.

LaSC has conceived within the scope of a Tripartite Partnership Agreement signed between ITAIPU Binacional, the Brazilian Army Command, through its Department of Science and Technology, and the ITAIPU-Brasil Technological Park Foundation.

3.1 LaSC Structure

The Cybernetic Security Laboratory of Cyber-physical Systems (LaSC) has a divided structure according to the ANSI/ISA-95 technical standard. In composition, the model is composed as shown in **Fig. 7.**



Fig. 7. LaSC general conception (Source: The Authors, 2022)

LaSC has an application server for developing the graphical interface and programming objects through OpenCIM Manager. Also, a database server, ethernet, USB, and serial data communication networks for integrating the communication between the mechanisms that make up the structure RFIDs, infrared sensors, precision scanners, programmable logic controllers (PLCs), robotic arms, and Human Machine Interface (HMI). The HMI, or computers for industrial use as they are also known, aims to integrate man's communication with the cyber-physical system, making production intelligent.

In addition, as a preventive action to simulate the production environment, the laboratory has intrusion sensors, which guarantee the safety of students and professionals on the production line.

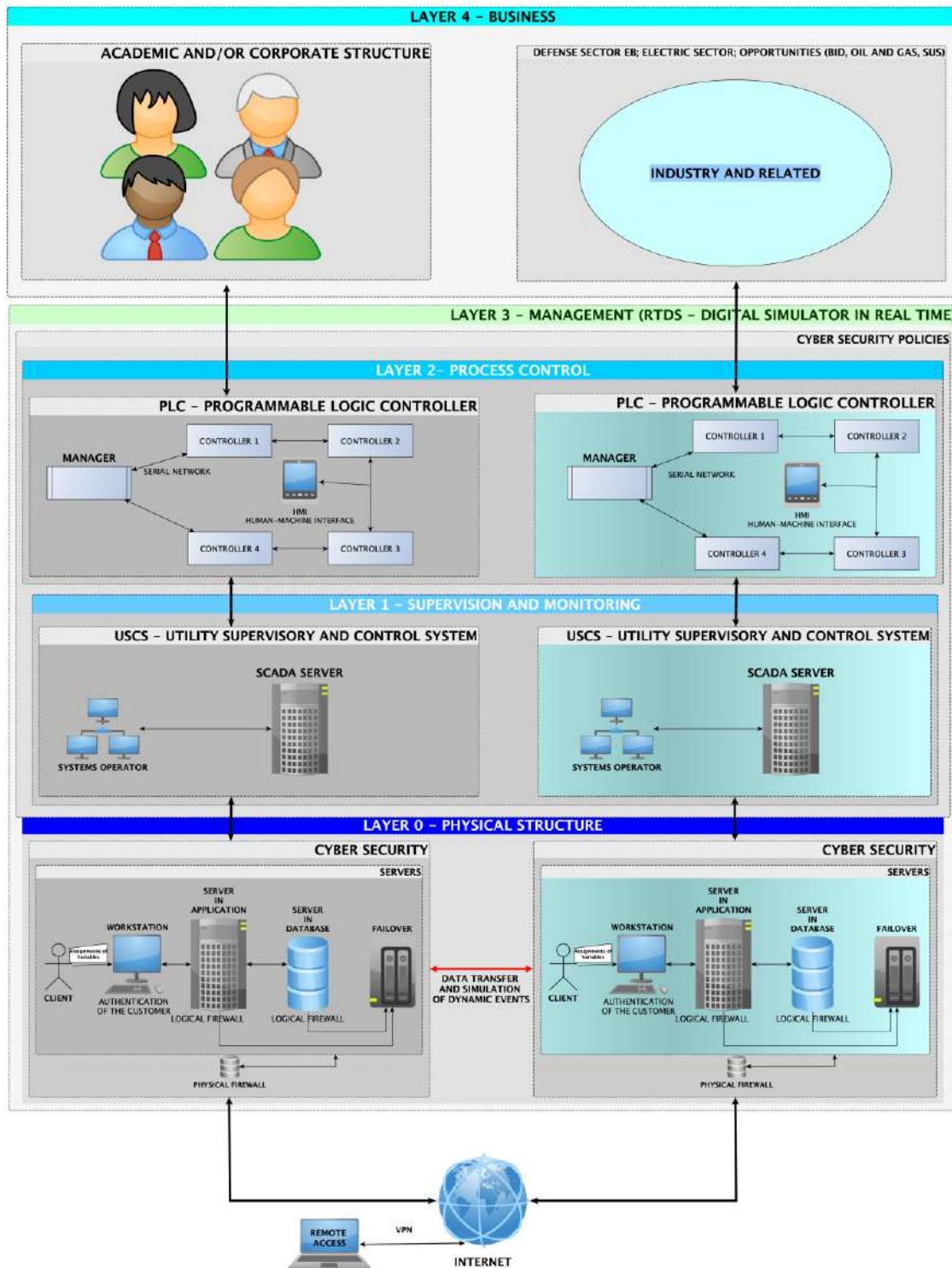


Fig. 8. Structure of a laboratory, based on LaSC (Source: The Authors, 2022)

According to ISA-95, which structures the organization of automation internationally, the Digital Twin interoperability layers are presented, considering the viability of the physical skeleton of a laboratory concomitant with industrial construction. That is, any project based on the concepts used in the composition of this research and its development regarding cybersecurity standards and the use of information security policies. In layer 0, the physical structure is where the data transfer and simulation of dynamic events occur. Layer one is the supervision and monitoring method, layer two is the control of processes, in layer three is the set of all meshing for management training, which establishes the execution of the concepts gathered in the development of the project. In addition to cybersecurity composition, to protect the database at layer 0,

there is also the Failover server, a redundant server for application and the database, to guarantee the production of large businesses. **Fig. 8** illustrates this arrangement.

Based on the structure of LaSC, it is possible to describe the behavior and performance of the laboratory, with emphasis on the main functionalities. A use case can be viewed and understood as a system utility performed through the interaction between the system and its actors [17]. The Use Case diagram describes how the actors use the system to reach a goal. **Fig. 9** illustrates each use case, represented by an ellipse, and describes a system behavior with interactions between actors. The user "Partner Institutions" in **Fig. 9** represents the external entities located in **Fig. 11**, namely: "sector of research, development, and innovation," "defense sector of the Brazilian Army," "opportunities," and "electrical sector."

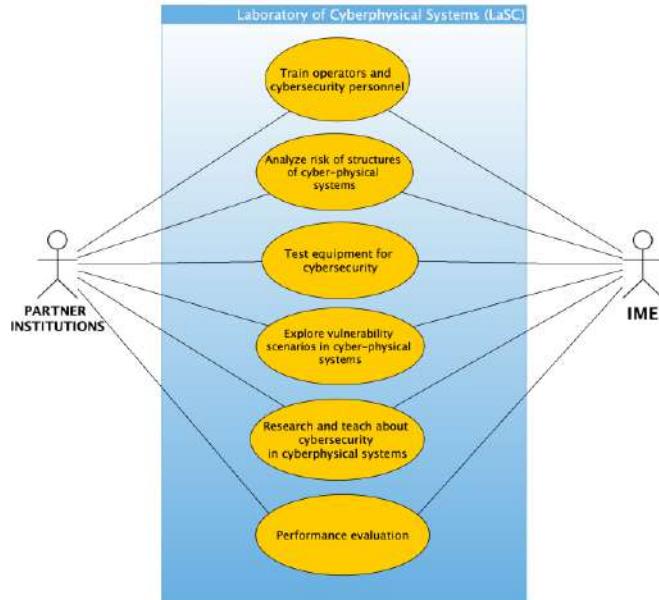


Fig. 9. LaSC Use Cases Diagram (Source: The Authors, 2022)

Exploring the use case "testing equipment for cybersecurity" shown in **Fig. 9**, it is possible to trace a flow of activities oriented towards studying and testing electrical systems in cybernetic environments. **Fig. 10** shows a real-time hardware-in-the-loop test and simulation plan of a stabilizer for an electrical power system (PSS – Power System Stabilizer). The PSS acts to mitigate unwanted electromechanical oscillations in an electrical power system, making the system more stable. The tests aim to evaluate the contribution of the PSS to the safety, performance, and integrity of the electrical power system.

Based on the demands, the requirements are established, and the electric power system model is designed. The functions of the electrical power system are reproduced through representation and modeling in software, such as RSCAD, which operates together with the RTDS hardware. Before the hardware-in-the-loop simulation, the modeling of the electric system is made together with the control system programming, in this case, the PSS. As the stabilizer attenuates power and frequency variations, it is necessary to program a frequency reading system and a voltage reading interface to verify the correct performance of the stabilizer. In this first stage of the simulation plan, the PSS is non-commercial hardware used in benchtop testing.

Next, a commercial PSS is programmed, once this PSS will be installed in the real plant of the electrical power system. Non-commercial hardware PSS programming can be done using software such as LabView together with RSCAD and RTDS. In the implementation of the commercial PSS that will be installed in the real plant, the programming depends on software compatible with the hardware of the commercial PSS. After modeling the real plant in RSCAD and programming the control system based on the commercial equipment that will be installed in the real plant, the electrical power system operating parameters are entered to perform a closed-loop test with the RTDS.

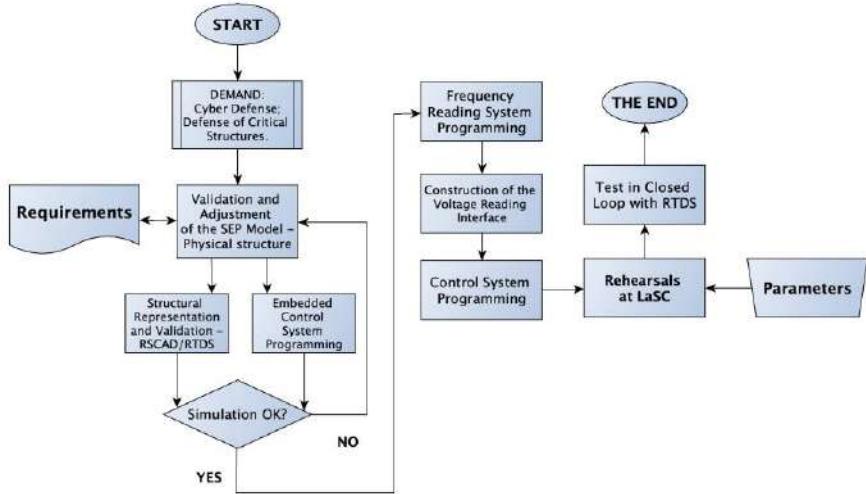


Fig. 10. LaSC Activities Diagram (Source: The Authors, 2022)

The organization of LaSC in **Fig. 11** uses technological capabilities. According to [18] technological capability is using specialized knowledge effectively. This ability to use technical knowledge and skills not only in the improvement and development of products and processes but also in the improvement of existing technologies, in addition to generating new knowledge and skills in response to the competitive business environment. [19]

Thus, it is possible to say that technological capacity is a stock of resources based on specialized knowledge, segmented into at least four components: technical and physical systems (physical capital), expertise and qualification of people (human capital), organizational system (organizational capital), and products and services, which represent the most visible part of technological capacity. [20]

According to the structure noted throughout the research, it is possible to see in the diagram of **Fig. 11** that the laboratory comprises a set of technological resources and utilities to be explored by Educational Institutions, governmental organizations, and private companies as a laboratory. It also presents the Defense sectors of the Brazilian Army that interact with the multidisciplinary capacity of related exchanges, business opportunities, and intellectual development.

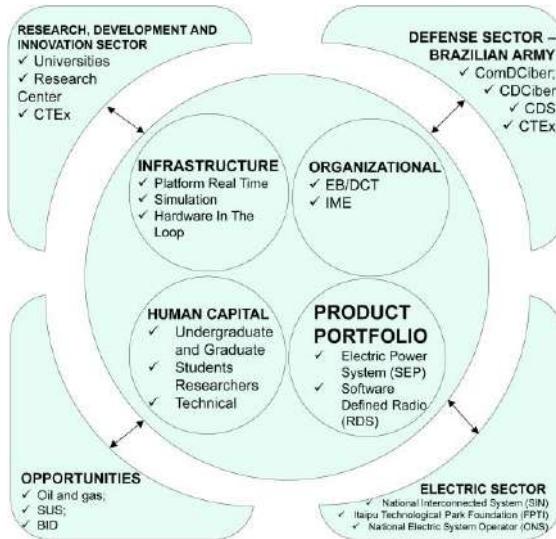


Fig. 11. Context Diagram outline (Source: The Authors, 2022)

3.2 Simulation of Cyber-physical Events

The concepts of digital twin and Cyber-Physical Power System (CPPS) are relatively new, and about which much is said within the context of digital transformation in industries. The sectors that involve large critical infrastructures have benefited from the creation of Cyber-Physical Power System, such as the oil

and gas industry and electric power systems (generation, transmission, and distribution), among others. Among the significant challenges of CPPS technology, we can mention the urgent need to protect data against threats and attacks. After all, cyber connectivity broadens the vulnerability surface of devices induced by network connectivity characteristics, for instance.

In a large-scale Cyber-Physical Power System (CPPS), a cyberattack alters the information flow in the cyber system, jeopardizing the secure operation of the physical power system. This may occur in a cascading failure throughout the CPPS.

The function of cybersecurity for monitoring, protection, and control purposes in any organization with industrial secrets, is to promote preventive actions such as simulating cybernetic events and building defense mechanisms against attacks. In Brazil, the Cyber Defense Command (ComDCiber), a military organization subordinated to the Brazilian Army, organizes the Cyber Guardian Exercise (CGE) every year. The CGE is a mega simulation of attacks on strategic National Defense infrastructures such as energy, communication, and transport, among other sectors. Through the simulation of cybernetic events, integrated response strategies are developed among divisions to combat possible cybernetic threats. This action, through collaborative action among the Armed Forces and civilians from academia and private companies, develops simulation techniques to protect against cyber threats.

It's been estimated that 63 different forms of advanced cyberattacks are being developed for CPPSs to attack monitoring, protection, and control functions.

LaSC focuses on studies and analyses in the field of cyber security of critical industrial infrastructures subjected to real malware attacks. An example of an electrical power system (EPS) comprises two environments, one electrical and the other cybernetic, thus representing a cyber-physical system. Both simulated systems are expected to be virtually connected through an encrypted internet connection to transfer simulated variables related to the electromechanical dynamics of the plants in real-time, creating a co-simulation environment. In summary, the LaSC enables simulations in controlled environments to test electrical and cybernetic equipment, allowing operators to practice processes in the real world that do not bring risks to people, equipment, or plants in general. This capability represents savings and safety for the company and the operators, greater mastery and ability to operate the systems, supporting the growing and complex decision-making quickly and accurately, and preserving the safety of the industrial processes when facing cyber threats.

Therefore, it is suggested for a simulation of cybernetic events the machine virtualization in various operating systems to evaluate the damage caused by malicious agents associated with different hardware and network architectures, applications, and communications protocols. Thus, in a virtual environment for high-stress testing with attacks and external interactions, without dismissing the internal ones, those generated by actors in loco, it will be possible during the simulation to observe in an advanced way, the behavior of malware, ransomware, and data exfiltration, enabling the development of defense mechanisms and protection preventive actions.

4 Conclusion

This work aimed to carry out an experimental project of the Digital Twin technology in cyber-physical systems. Some SysML and UML diagrams guided the preparation of this article. The diagrams are a great strategy to give the direction to be followed during the work.

For the academic part, on the study content involved, the diagrams listed were Use Cases, Block Diagrams, Context Diagrams, and Activities Diagrams. With these diagrams, it is possible to elucidate in an easy way to understand the operation of the technology used within the Laboratory of Cyber-physical Systems (LaSC).

Given the making of the diagrams for both the management of the exploratory project and the elucidation of the complex content involved, the initially proposed objective, the realization of an exploratory project of the Digital Twin technology in cyber-physical systems, was achieved.

In future works, it would be great to create more detailed diagrams of use cases, showing the interactions carried out internally by the components of the laboratory. It will also be good to create a sequence diagram for project implementation showing more details, like the dynamics between the demands requested from the laboratory and the way to meet such needs. Furthermore, as a future research proposal, the metaverse could be explored as an additional virtual world theme in Digital Twin technology.

As technology advances over the years, complex systems are becoming more and more the lived reality of each day. That is why it is necessary to continuously study the methodologies used to evaluate, test, and improve the increasingly common complex systems.

References

1. Cardoso, R. C. M.: Caminhos da Manufatura - Uma Abordagem à Manufatura Digital. [S.l.]: eBook Kindle (2016)
2. Alam, M.; Saddik, A.: C2PS: A Digital Twin Architecture Reference Model for the Cloud-Based Cyber-Physical Systems. IEEE Access, v. 5, pp. 2050-2062 (2017)
3. ISO; IEC; IEEE. ISO/IEC/IEEE 15288:2015 Systems and Software Engineering - System Life Cycle Processes. Genebra (2015)
4. INCOSE, <https://www.incose.org/>
5. Abordagem: Engenharia de Sistemas na Concepção, Implementação e Gestão dos Projetos de Sistemas de Engenharia (Matemática, Esboços, Planejamento), <https://sites.google.com/ime.eb.br/engsistemas/turma2021/abordagens-esboço-matemática-planejamento-e-gestão>
6. Iyer, A.: Digital Twin: Possibilities of The New Digital Twin Technology. Edição do Kindle (2017)
7. Siemens, <https://www.siemens.com>
8. Siebel, T. M.: Transformação Digital. Rio de Janeiro: Editora Alta Books, 2021. E-book. ISBN 9788550816876 (2021)
9. Almeida, P. S. D.: Indústria 4.0 - Princípios Básicos, Aplicabilidade e Implantação na Área Industrial. São Paulo: Editora Saraiva, 2019. E-book. ISBN 9788536530451 (2019)
10. Masaro, L., https://files.cercomp.ufg.br/weby/up/410/o/Disserta%C3%A7%C3%A3o_-_Cibernetica_Ciencia_e_T%C3%A9cnica.pdf
11. Dupuy, J. P.: Nas Origens das Ciências Cognitivas. Tradução Roberto Leal Ferreira. Editora UNESP. São Paulo (1995)
12. Wiener, N.: Cybernetics: or Control and Communication in the Animal and the Machine. Cambridge, Massachusetts. MIT Press. 1^a Ed: 1948. 2^a Ed: (1961)
13. Chaves, V. H. C.: A Revolução Cibernetica: A Nova Cultura. XIX EBRAPEM (2015)
14. Lobão, E. C., Porto, A. J. V.: Evolução das Técnicas de Simulação. PRODUÇÃO, Vol 9, nº1, pp. 13-22, ABEPROM, Rio de Janeiro (1999)
15. Rueda, O. A. S.: Simulador em Tempo Real Baseado na Integração de Módulo FPGA e Cpus para Avaliação de Controladores Embarcados de Conversores Eletrônicos de Potência. UFRJ. Rio de Janeiro (2019)
16. Faruque, M. D. O.: et al. Real-Time Simulation Technologies for Power Systems Design, Testing, and Analysis. IEEE Power and Energy Technology Systems Journal, vol. 2, no. 2, 63-73, June 2015, DOI: 10.1109/JPETS.2015.2427370 (2015)
17. OMG Systems Modeling Language, <https://sysml.org/.res/docs/specs/OMGSysML-v1.4-15-06-03.pdf>
18. Kim, L.: Building Technological Capability for Industrialization: Analytical Frameworks and Korea's Experience. Oxford, v.8, n.1, Mar (1999)
19. Jin, J., Von, Z. M.: Technological Capability Development in China's Mobile Phone Industry. Technovation, v.28, n.6, pp.327-334 (2008).
20. Figueiredo, P. N.: Gestão da Inovação: Conceitos, Métricas e Experiências de Empresas No Brasil. 2ed. Rio de Janeiro: LTC (2015)

ABSTRACTIVE TEXT SUMMARIZATION FOR TAMIL LANGUAGE USING m-T5

C. Saraswathi^{1*}, V.Prinitha², and J. Briskilal³

Department of Computing Technologies
SRM Institute of Science and Technology
Kattankulathur, Chengalpattu – 603203.
¹sc8272@srmist.edu.in, ²pv6929@srmist.edu.in,³briskilj@srmist.edu.in

Abstract Summarization is the act of reducing a long form of text to a shorter version by lowering the length of the original text while retaining and keeping the summary's core and the informational parts which conveys the content's meaning. Summarizing lengthy papers by hand is difficult and prone to inaccuracy. The exponential rise of information and data has made Automatic Text Summarization crucial. It leads to a summary generated by a summarizing system that enables users to understand the document's contents. Abstractive and Extractive Summarization are the two categories into which summarization falls. We can see that there aren't many automatic text summarizers available for Indian languages. Our main goal in this direction is to create an automatic abstractive text summarizer for Tamil using the m-T5 Transformer model. We have also used manual dataset of Tamil Kid's stories for training and testing.

Keywords: Automatic Text Summarization, Abstractive Summarization, Transformer model, m-T5 model, Tamil language.

1. Introduction

The amount of textual content has significantly increased in recent years, making it a fantastic resource for information extraction and analysis from diverse sources. It is necessary to summarize the information in order to quickly identify important information from a huge text content. Text summarization has emerged as one of the most crucial and useful tools among all contemporary technologies enabling consumers to quickly comprehend a massive amount of information. The task of condensing a lengthy text's contents into a manageable length is known as summarization. Summarization is mostly used to condense the original text while maintaining the essential informational components and meaning of the content. Automatic Text Summarization system, one of the specialized data mining tools, aids in this work by providing a brief summary of the material because manual summarization of documents frequently involves errors. The exponential explosion of information and data has made Automatic Text Summarization a significant boon. There are many different summarizing methods available in natural language processing. Abstractive Text Summarization is a challenging problem in natural

language processing. Furthermore, little research has been conducted in regional tongues. In contrast to other summarization techniques like the extractive one that reuses the words and phrases from the original text, the abstractive text summarization method creates a succinct and to-the-point summary of a lengthy text document. Here, none of the words or sentences from the original paper are used.

Till now no pre-existing research works have been done for Abstractive Text Summarization for Tamil language. Only Extractive Approaches like Semantic Graph Method and Centroid Approaches are available for Tamil language which are more human like rather than algorithmic and robotic. This is because they merely take high priority sentences from the source text that is depicted in the summary instead of creating new terms. This means that sentence coherence, consecutive sentence relationships, and paragraph structure are not taken into account, which gives the summary a robotic form. Our suggested effort aims to create an easily understandable abstractive summary of a Tamil Document. Tamil is a Dravidian language primarily spoken in India. Tamil Nadu, an Indian state, as well as Pondicherry, a union territory, both have Tamil as their official language. There are sizable populations of speakers in Fiji, Malaysia, South Africa, and Mauritius, and it is also an official language in Sri Lanka and Singapore. Two things set Tamil apart from many other languages: 1) Tamil lacks articles. 2) Tamil is a null subject language. Not every Tamil sentence have a subject, verb, and object. In Tamil, it is also feasible to create meaningful sentences using simply a verb. Therefore, the foundation of our project is creating an automatic text summarizer for Tamil.

Here for example we have taken a tamil story:

சக்கரவர்த்தி அக்பர் திடீரென உணர்ச்சிவசப்படுவார்! அவர் எப்போது உற்சாகத்துடன் இருப்பார், எப்போது எரிந்து விழுவார் என்று சொல்ல முடியாது. பல சமயங்களில், அவரையே கேலி செய்வதுபோல் அமைந்துள்ள நகைச்சவைத் துணுக்குகளைக் கேட்டும் கோபம் அடையாமல் சிரிக்கவும் செய்வார். ஆனால், சில சமயங்களில் சில சாதாரண துணுக்குகள் கூட அவரை கோபமுறச் செய்யும்.

தர்பாரிலுள்ள அனைவருக்கும் அக்பரின் அடிக்கடி மாறுபடும் மனநிலையைப் பற்றித் தெரியும். அக்பரின் பேகத்திற்கும் இது தெரியும். ஆனால், பேகம் அதைப் பொருட்படுத்தாமல் பல சமயங்களில் அவரைக் கிண்டல் செய்வதுண்டு. சிலசமயம் அவனுடைய கேலிப் பேச்சினால் கோபமடைந்தாலும், சக்கரவர்த்தி உடனே அவளிடம் சாந்தமாகி விடுவார் என்ற அனுபவம்தான் காரணம்!

ஓருநாள் மாலை நேரம், அக்பரும் பேகமும் அந்தப்புரத்தில் அருகருகே அமர்ந்திருந்தனர். சாளரத்தின் வழியே வீசிய தென்றல் காற்றில் மல்லிகை மணம் தவழ்ந்தது. அதை அக்பர் வெகுவாக ரசித்துக் கொண்டிருந்தார். அப்போது அவரைக் கிண்டல் செய்ய வேண்டுமென்று பேகத்திற்குத் தோன்றியது. “ஏது! மல்லிகை மணம் உங்களை மயக்குகிறதோ? என்னிடம்

இல்லாதது மல்லிகையில் அப்படி என்ன இருக்கிறது?" என்று வாயைக் கிண்டினாள்.

The Given Text is Summarized as:

சக்கரவர்த்தி அக்பர் தீஸர் திமிரென உணர்ச்சிவசப்படுபவர். அவர் எப்போது உற்சாகத்துடன் இருப்பார், எப்போது எரிந்து விழுவார் என்று சொல்லமுடியாது. தர்பாரிலுள்ள அனைவருக்கும் மற்றும் அக்பரின் பேகத்திற்கும் இது தெரியும். ஆனால், பேகம் அதைப் பொருட்படுத்தாமல் பல சமயங்களில் அவரைக் கிண்டல் செய்வதுண்டு. ஒருநாள் மாலை நேரம், அக்பரும் பேகமும் அந்தப்புரத்தில் அருகருகே அமர்ந்திருந்தனர்சாளரத்தின். வழியே வீசிய தென்றல் காற்றில் மல்லிகை மணம் தவழ்ந்ததுஅதை. அக்பர் வெகுவாக ரசித்துக் கொண்டிருந்தார் அப்போது. அவரைக் கிண்டல் செய்ய வேண்டுமென்று பேகத்திற்குத் தோன்றி யதுமல்லிகை. மணம் உங்களை மயக்குகிறதோ?என்னிடம் இல்லாதது மல்லிகையில் அப்படி என்ன இருக்கிறது?"என்று வாயைக் கிண்டினாள். அதற்கு அக்பர் ஆம்மல்லிகை! மணம் என்னை மயக்குகிறதுஅதிலுள்ள. மயக்கம் உன்னிடம் இல்லை!" என்றார்

2. Literature Survey

Extraction and abstraction are the two basic divisions of automatic text synthesis methods. Extractive methods produce summaries by extracting phrases and words from the source text. Statistical, conceptual, optimizational, topical, graphical, sentence centrality, semantic, and deep learning-based approaches are different types of methodology.

[1] Eduardo Fidalgo and Akanksha Joshi made a proposal Ranksum, As part of a method for obtaining text summaries from single documents, four multidimensional sentence aspects—subject information, semantic content, significant keywords, and position—are recovered for each phrase. They have used numerous extractor types to rank sentences. LDA (Latent Dirichlet Allocation) was used in Topic rank extractor to find the topic vectors using the training dataset.Sentence embeddings are utilised in Embedding-based To locate meaningful sentences based on their semantics, use Semantic Rank Extractor-Next. They have used SBERT. In order to determine the

important phrases in a document, keyword rank extractor uses lemmatization to compute the set of keywords. Position rank extractors produce position rank vectors by assigning phrases a rank depending on their position. A weighted rank fusion from all four derived ranks is calculated.

[2] Jintao Zhao and Libin Yang proposed the Heterogeneous tree structure-based extractive Summarization(HetreeSum) model,which learns the links between phrases by representing each text as a tree structure.The tree structure also includes the structural information from the original document, giving it a broad view of the entire content.First, a heterogeneous document tree is built (where each word is represented as a leaf node of the tree structure, sentences and sections as branch nodes of the tree structure). The three components that make up the heterogeneous document tree are the Tree Initializer, TreeTransformer Layer, and Sentence Selector.

[3] YongpingDua and QingxiaoLi introduced a novel model called BioBERTSumwhich has been introduced to, bettercapture token-level and sentence-level contextual representation. It employs a domain-aware bidirectional language model that has been further refined for extractive text summarization tasks on a single biomedical document. This language model has been pre-trained on large biomedical corpora.

[4] PradeepikaVerma and AnshulVerma described An two-stage hybrid methodology for text summarization that combines the advantages of various methodologies.Using a partitional clustering technique, they categorise the sentences in a document according to how similar they are in the first stage. Then, to distinguish between two texts, they employ a linear combination of the normalised Google distance and the word mover's distance. To find keywords that are semantically similar, they use the Normalized Google Distance (NGD) semantic similarity metric. The significant phrases are then extracted from each cluster based on the revised text feature scores from the second step (partition). The best weights for the sentence are currently being determined using a teaching-learning-based optimization technique. The significant sentences are recovered using the Optimal Weights. A Fuzzy Interference System with a human-created knowledge base is also used to calculate the sentence's final score.

[5] MouradOussalah and Muhidin Mohamed proposed an approach for text summarization called SRL-ESA-TextSum, which is based on semantic role labelling and explicit semantic analysis. A unique framework for the text summary of single and multiple texts using graphs is provided in this research. Using SRL, each word having semantic roles is recognised by its corresponding semantic Role Tags. In phrase level semantic parsing, each sentence is represented as a multi-node vertex that contains Wikipedia concepts for semantic arguments. The summarization system is evaluated using the standard publicly accessible dataset from the 2002 Document Understanding Conference (DUC 2002).

[6]Alexander M. Rush, Sumit Chopra, JasonWeston proposed A Neural Attention Model for Abstractive Sentence Summarization.Abstactive techniques are used to create summaries by condensing phrases into shorter terms and using words that do not appear in the original text. Even though extractive summarizing of phrases has

generated a sizable body of study, Abstractive summarization has gotten much less attention. For the first time in 2015, Rush et al. applied deep learning to abstractive text summarization. The Rush et al. model used three different types of encoders: convolution, bag-of-words, and attention-based encoders. Deep learning algorithms have recently been frequently used in abstractive text summarization due to their positive outcomes.

[7] Dima Suleiman and Arafat Awajan proposed an abstractive method for summarizing Arabic text and it primarily uses the sequence-to-sequence RNN architecture. It is made up of a multilayer encoder and a single-layer decoder. The decoder builds the summary words using a global attention method that takes into account all of the input hidden states. In addition to ROUGE 1, three additional evaluation metrics, known as ROUGE1-NOORDER, ROUGE1-STEM, and ROUGE1-CONTEXT, are recommended.

[8] Baykara and Tunga Gungor proposed an abstractive text summarization for the Turkish and Hungarian languages, they created two significant datasets, TR-News and HU-News. Both the pointer-generator model and the BERT+Transformer model have been applied to text summarization in this work. The pointer-generator model has been chosen as the default model. An encoder-decoder network can choose whether to highlight a word from the input sequence or to produce a new term from the vocabulary. The decoder is a unidirectional LSTM with attention mechanism, while the encoder is a bidirectional LSTM. Their encoder-decoder system, known as BERT + Transformer, uses BERT as the encoder and a 6-layered transformer network as the decoder. Because meaning in these agglutinative languages is mostly held within the morphemes of the words, morphology is crucial. Modern Abstractive summarization models are utilised to assess how well the proposed linguistically-focused tokenization algorithms (Separate Suffix and Combined Suffix) function. The two techniques used in this study are separate suffix and combined suffix, and both are based on the roots of the nouns and their suffixes. On the TR-News and HUNews datasets, they discovered that the multilingual cased BERT model beat the monolingual BERT models (BERTurk and huBERT) in terms of ROUGE score. On the TR-News dataset, the Separate Suffix approach produces the greatest ROUGE-1 score, and the HU-News dataset shows promising results.

[9] XiaoyanCai and Kaile Shi created an attentional neural model based on HITS that treats the sentences and words in the original manuscript as hubs and authorities in order to properly utilise sentence-level and word-level information. The sentence and words with the highest likelihood are selected throughout this process after the sentences' attention values are assessed using Kullback-Leibler divergence. KL Divergence allows us to quantify the amount of information we lose when selecting an approximation. They have proposed a comparison method using the Hierarchical Beam Search Algorithm to further improve the summarization performance.

[10] XiaoyanCai and Sen Liu proposed COVIDSum, a SciBERT-based summarization technique for COVID-19 scientific papers. On the basis of the chosen sentences, co-occurrence graphs are constructed to depict the linguistic features of these phrases. They used CORD-19 heuristic techniques to extract pertinent sentences from the source articles. Following that, the sentences and word co-occurrence graphs

were encoded using SciBERT and a graph encoder based on a graph attention network (GAT), respectively. Highway networks, which also incorporated linguistic knowledge into the contextual embeddings of scientific publications, were subsequently used to combine the two encodings mentioned above. Then, for each scientific paper, an abstractive summary was produced using a transformer decoder. Multiple heads of attention, a point-wise feed-forward layer, residual connections, and other features are present in the Transformer decoder. At the t-th timestep, this layer ground-truth word should be expected.

2.1 Existing System and Novelty :

For Tamil language only extractive text summarizations techniques are available.Till now there are no research works done on Tamil Abstractive Text Summarization. Tamil Summarization using Extractive Approaches like Semantic Graph and Centroid Approach generate summaries that are more robotic and algorithmic rather than human like. This is because they merely take high priority sentences from the source text that is depicted in the summary instead of creating new terms.This means that sentence coherence, consecutive sentence relationships, and paragraph structure are not taken into account, which gives the summary a robotic form.In Centroid Approach, Each sentence within a document is treated as a vector in a multidimensional space.The sentences that are closest to the value of the centroid are regarded the most important[11]. LF Parser is used in the Semantic Graph Approach to identify the text's semantic properties. Coreference, feature extraction, and named entity Resolution is carried out as much as is required for summarization[12]. So we have proposed a method of Abstractive Summarization for Tamil language using m-T5.We have created an Inhouse dataset named Tamil kids stories which contains 100 Tamil kids stories. The proposed dataset has title, story and reference summary in it. Each story approximately contains 854 words and 772 words in reference summary. These stories are scrapped from Tamil article site and the reference summary has been manually created for each story in order to get an efficient output and for the better performance of m-T5 model.

3. Proposed Work

Our idea is based on a field of artificial intelligence known as natural language processing, which enables machines to read, comprehend, pick up on, and derive meaning from human language. Using m-T5, we have proposed an abstractive text summarizing method for the Tamil language. Using a novel design known as NLP, this m-T5 Transformer model promises to solve sequence-to-sequence issues while deftly addressing long-distance dependencies. m-T5 is a multilingual version of Google's T5 model that has between 300 million and 13 billion parameters and was pre-trained using a dataset of 101 languages[13]. The m-T5 model's architecture and

training were closely modelled by those of the T5 model. To train mT5, we introduce a multilingual variant of the C4 dataset called mC4. The massive clean crawled corpus (mC4), comprises natural text in 101 languages drawn from the public Common Crawl web scrape.

T5 is short for "Text-to-Text Transfer Transformer". As the name implies, it is an encoder-decoder paradigm that uses transformers for text production[14].

Similar to the Transformer model[15], the T5 model has a full encoder-decoder design. The job of the encoder is to convert the input from a series of discrete representations to a series of continuous representations, which is then supplied into a decoder. Transformer architecture does not use recursion, therefore it cannot by nature capture any information about the relative placements of words in the sequence. By adding positional encodings to the input embeddings, this data must be injected.

Prior to being fed into the neural network, the input is first delivered to positional encoding, which assigns numbers to each word in a sentence based on its position in the phrase. The network then learns how to decode such positional encodings as we train it on a large amount of text input.

Encoder Stack:

The encoder consists of a stack of $N = 6$ identical layers. Each layer has two sub-layers. The multi-head self-attention technique is implemented in the first sublayer. In general, paying attention to all the other words in context is the only way to determine a word's real meaning (representation). One method for doing that is through self-attention. In order to decide how to translate a word in the output, the text model can now examine every word in the source phrase. The purpose of this is to provide the attention function access to information that would otherwise be inaccessible to it when using a single attention head.

An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The input consists of queries and keys of dimension d_k , and values of dimension d_v . To determine the weights on the values, we compute the dot products of the query with each key, divide them by d_k , and then use the softmax algorithm.

The input is made of queries and keys of dimension d_k , and values of dimension d_v . The dot products of the query is computed with all keys, divided each by $\sqrt{d_k}$, and apply a softmax function to obtain the weights on the values. In practice, we simultaneously compute the attention function on a set of queries that are grouped into a matrix Q .

The keys and values are also packed together into matrices K and V . The matrix of outputs is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T/\sqrt{d_k})V \quad (1)$$

It was found useful to linearly project the queries, keys and values h times with different, learned linear projections to d_q , d_k and d_v dimensions, respectively.

The projected copies of the queries, keys, and values are then each simultaneously

subjected to the attention function to provide d_v -dimensional output values. The final values are created by concatenating these and then projecting them once more.

Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions. With a single attention head, averaging inhibits this.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

$$\text{where } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{h d_v \times d_{\text{model}}}$.

In this work we employ $h = 8$ parallel attention layers, or heads. For each of these we use $d_k = d_v = d_{\text{model}}/h = 64$.

Layer normalization is applied to the input of each subcomponent. A simplified version of layer normalization is used where the activations are only rescaled and no additive bias is applied. After layer normalisation, each subcomponent's input and output are combined via a residual skip connection. Dropout is used at the input and output of the entire stack, on the skip connection, within the feed-forward network, and on the attention weights.

A feed-forward network with complete connectivity makes up the second sublayer. The information only goes in one direction in a feed-forward network, which serves this purpose. It is transmitted from the input nodes to the output node, passing through any hidden nodes (if any). This consists of two linear transformations with a ReLU activation in between.

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2(2)$$

The encoder sends the features to each block of the decoder. The encoder and decoder are comparable in many ways.

Decoder Stack:

Additionally, the decoder consists of a stack of six similar layers, each of which is made up of three lower layers: The information is delivered to the first sublayer, which then applies masked multi-head self-attention to it.

The decoder has been altered to focus just on the words that came before. In the first sublayer, masking is used to prevent the decoder from focusing on words that follow. A multi-head self-attention technique, identical to the one used in the first sublayer of the encoder, is implemented in the second layer. This multi-head mechanism gets the keys and values from the encoder's output as well as the queries from the previous decoder sublayer on the decoder side. As a result, the decoder can focus on every word in the input sequence. A fully connected feed-forward network, identical to the one used in the second sublayer of the encoder, is implemented in the third layer.

Finally, a fully connected layer, followed by a softmax layer, are applied to the decoder's output to produce a prediction for the following word in the output sequence.

Our model is trained on a variety of tasks that require it to create text after being provided text for conditioning or context. Our model was trained using Tamil kids's stories dataset which we have manually created using web scrapping techniques.

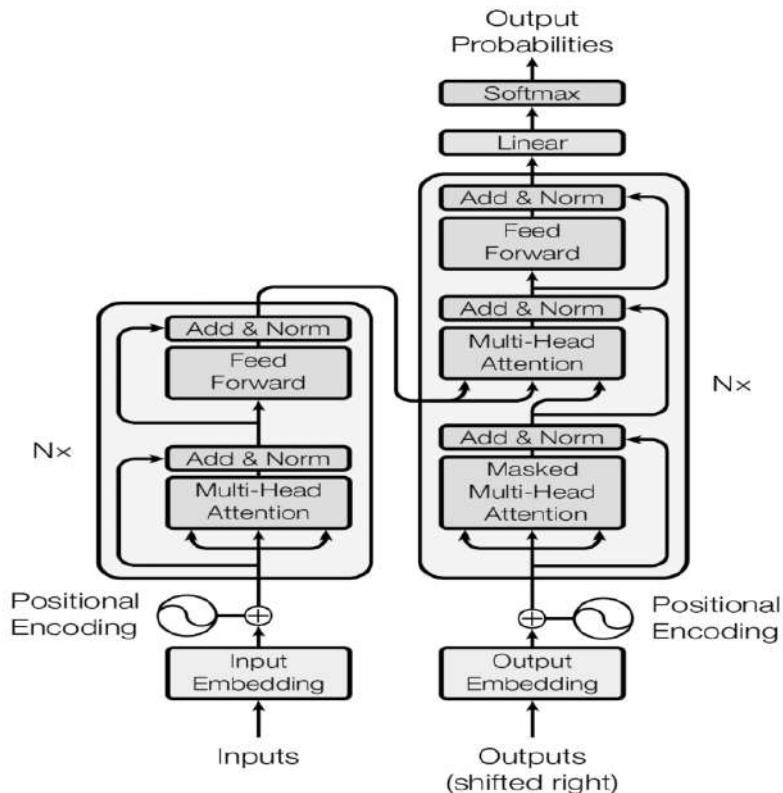


FIGURE 1 - Architecture of Transformer model

4. Implementation

T5 is short for "Text-To-Text Transfer Transformer". As the name implies, it is an encoder-decoder paradigm that uses transformers for text production.m-T5 is

multilingual version of t5. It has been pretrained using mC4 corpus of 101 languages. Similar to the Transformer model, the T5 model has a full encoder-decoder design. Our model was trained using Tamil Kids Stories dataset. There are 100 stories in the dataset. For validation purposes, 20% of the data are used. We have used 80% of data for training purposes. The m-T5 model has a maximum sequence length of 512, a batch size of 128 sequences, and has been pre-trained on C4 for $219 = 524,288$ steps. When fine-tuning, we employ a constant learning rate of 0.001. A batch size of 150 sequences and a maximum sequence length of 512 are used for fine tuning. Our models are trained for a single period. For our project, we're using the Adam optimizer. We have used Beam Search Decoding method. By maintaining the most likely num beams of hypotheses at each time step and eventually selecting the hypothesis with the highest overall probability, beam search lowers the danger of missing hidden high probability word sequences. To ensure that generation is complete when all beam hypotheses have reached the EOS token, we set num beams = 2 and early stopping = True.

5. Result Discussion

Using the m-T5 model, Text Summarization is performed. We have employed the ROUGE scoring metrics. A Collection of Metrics and Software Programme called ROUGE, or Recall-Oriented Understudy for Gisting Evaluation, are used to assess automatic summarization and machine translation software in natural language processing. The metrics compare an automatically generated summary or translation with a reference summary or translation.

We have found out F1 Score, Recall and Precision of Rouge 1, Rouge 2 and Rouge L.

ROUGE 1:

ROUGE-1 precision can be computed as the ratio of the number of unigrams in generated summary that appear also in Reference Summary, over the number of unigrams in generated summary.

ROUGE-1 recall can be computed as the ratio of the number of unigrams in Reference Summary that appear also in generated summary, over the number of unigrams in Reference Summary.

ROUGE-1 F1-score can be directly obtained from the ROUGE-1 precision and recall using the standard F1-score formula.

We got rouge-1 recall, precision, F1-score as 0.4076, 0.78024 and 0.5219.

ROUGE 2:

ROUGE-2 precision is the ratio of the number of 2-grams in generated summary that appear also in reference summary, over the number of 2-grams in generated summary.

ROUGE-2 recall is the ratio of the number of 2-grams in reference summary that appear also in generated summary, over the number of 2-grams in reference summary.

$$F1 - score = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall}) \quad (3)$$

We got rouge-1 recall, precision, F1-score as 0.5551, 0.7393 and 0.6719.

ROUGE-L:

ROUGE-L is based on the longest common subsequence (LCS), which is the largest sequence of words (not necessarily consecutive, but still in order) that both our model output and reference share.

ROUGE-L precision is the ratio of the length of the LCS, over the number of unigrams in generated summary.

ROUGE-L recall is the ratio of the length of the LCS, over the number of unigrams in Reference Summary.

We got rouge-L recall, precision, F1-score as 0.4076, 0.7802 and 0.5219.

Table 1 represents the results of the proposed work

ROUGE SCORE	PRECISION	RECALL	F1 SCORE
ROUGE 1	0.78024	0.4076	0.5219
ROUGE 2	0.7393	0.5551	0.6719
ROUGE L	0.7802	0.4076	0.5219

TABLE 1: ROUGE SCORING FOR SUMMARY GENERATED

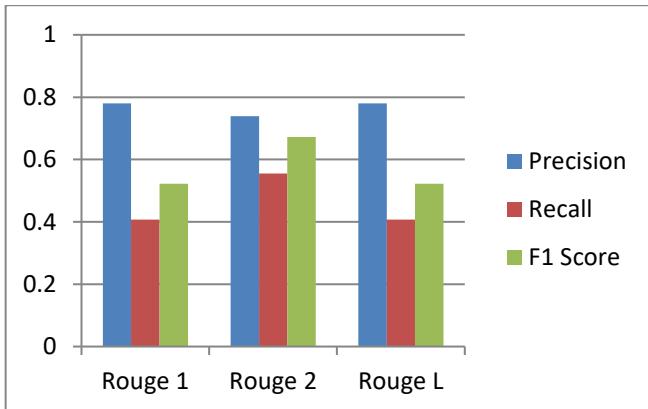


FIGURE 2 :BAR CHART REPRESENTING THE ROUGE SCORES

Example Predictions on Tamil kids story dataset:

To show that our model is generating fluent summaries, we include a few example decodes from our model on the validation set along with the actual summary. Here by we have given below few lines of a single story.

Actual Summary:

சக்கரவர்த்தி அக்பர் திடீரென உணர்ச்சிவசப்படுபவர். அவர் எப்போது உற்சாகத்துடன் இருப்பார், எப்போது ஏரிந்து விழுவார் என்று சொல்ல முடியாது. தர்பாரிலுள்ள அனைவருக்கும் மற்றும் அக்பரின் பேகத்திற்கும் இது தெரியும். ஆனால், பேகம் அதைப் பொருட்படுத்தாமல் பல சமயங்களில் அவரைக் கிண்டல் செய்வதுண்டு. ஒருநாள் மாலை நேரம், அக்பரும் பேகமும் அந்தப்புரத்தில் அருகருகே அமர்ந்திருந்தனர்சானரத்தின். வழியே வீசிய தென்றல் காற்றில் மல்லிகை மணம் தவழ்ந்ததுஅதை. அக்பர் வெகுவாக ரசித்துக் கொண்டிருந்தார் அப்போது. அவரைக் கிண்டல் செய்ய வேண்டுமென்று பேகத்திற்குத் தோன்றியதுமல்லிகை. மணம் உங்களை மயக்குகிறதோ?என்னிடம் இல்லாதது மல்லிகையில் அப்படி என்ன இருக்கிறது?” என்று வாயைக் கிண்டினான். அதற்கு அக்பர் ஆம்மல்லிகை! மணம் என்னை மயக்குகிறதுஅதிலுள்ள. மயக்கம் உண்ணிடம் இல்லை!” என்றார் திடீரென எரிச்சலுற்ற. மனைவி என்ற முறையில் உங்களிடமிருந்து எதையும் எதிர்பார்க்க எனக்கு உரிமை உண்டு!” என்று பதிலளித்தார் பேகம்.“அந்த உரிமை யாருக்கும் கிடையாதுநான். மற்றவர்களைப் போல் சாதாரண மனி தனில்லைனனில். நான் சக்கரவர்த்தி!” என்று உரக்க முழங்கினார் அக்பர். எனக்குக்கூட கிடையாதா?நான் என்ன சாதாரணப் பெண்ணா?” என்றார் பேகம். அக்பர் தான் கேவியாகக் கேட்டதில் கோபமடைந்து விட்டார்

என்று உணர்ந்தால். அந்த ஸ்தானத்தை நீ இழக்கும் வேளை நெருங்கிவி ட்டது!” என்று அக்பர் . அதை சற்றும் எதிர்பாராத பேகம் அழுதே விட்டான்.

GENERATED SUMMARY:

சக்கரவர்த்தி அக்பர் திமீர் திமீரென உணர்ச்சிவசப்படுவர்அவர்! எப்போது உற்சாகத்துடன் இருப்பார், எப்போது எரிந்து விழுவார் என்று சொல்ல முடியாதுபல. சமயங்களில், அவரையே கேலி செய்வதுபோல் அமைந்துள்ள நகைச்சவைத் துணுக்குகளைக் கேட்டும் கோபம் அடையாமல் சிரிக்கவும் செய்வார்ஆனால்., சில சமயங்களில் சில சாதாரண துணுக்குகள் கூட அவரை கோபமுறச் செய்யும்தர்பாரிலுள்ள. அனைவருக்கும் அக்பரின் அடிக்கடி மாறுபடும் மனநிலையைப் பற்றித் தெரியும்அக்பரின். பேகத்திற்கும் இது தெரியும்ஆனால்., பேகம் அதைப் பொருட்படுத்தாமல் பல சமயங்களில் அவரைக் கிண்டல் செய்வதுண்டுசிலசமயம். அவனுடைய கேலிப் பேச்சினால் கோபமடைந்தாலும், சக்கரவர்த்தி உடனே அவளிடம் சாந்தமாகி விடுவார் என்ற அனுபவம்தான் காரணம்ஒருநாள்! மாலை நேரம், அக்பரும் பேகமும் அந்தப்புரத்தில் அருகருகே அமர்ந்திருந்தனர்சாளரத்தின். வழியே வீசிய தென்றல் காற்றில் மல்லிகை மணம் தவழ்ந்ததுஅதை. அக்பர் வெகுவாக ரசித்துக் கொண்டிருந்தார்அப்போது. அவரைக் கிண்டல் செய்ய வேண்டுமென்று பேகத்திற்குத் தோன்றியது.“ஏதுமல்லிகை! மணம் உங்களை மயக்குகிறதோ?என்னிடம் இல்லாதது மல்லிகையில் அப்படி என்ன இருக்கிறது? ” என்று வாயைக் கிண்டினாள்.“ஆம்மல்லிகை! மணம் என்னை மயக்குகிறதுஅதிலுள்ள. மயக்கம் உன்னிடம் இல்லை!” என்றார் திமீரென எரிச்சலுற்ற அக்பர்.“என்னைத்தான் நேசிக்கிறீர்கள் என்று நினைத்தேன்என்னைவிடமல்லிகை மீதுதான் மோகமா?” என்று பேகம் மீண்டும் வம்புக்கிழுத்தான். “ஆமாம்உன்! மீது மோகம் இருக்கவேண்டுமென்று என்ன அவசியம்?” என்றார் மேலும் கோபமுற்ற அக்பர்.“மனைவி என்ற முறையில் உங்களிடமிருந்து எதையும் எதிர்பார்க்க எனக்கு உரிமை உண்டு!” என்று பதிலளித்தார் பேகம்.“அந்த உரிமை யாருக்கும் கிடையாதுநான். மற்றவர்களைப் போல் சாதாரண மனிதனில்லைஞனில். நான் சக்கரவர்த்தி!” என்று உரக்கமுங்கினார் அக்பர்.“என்ன?எனக்குக்கூட கிடையாதா?நான் என்ன சாதாரணப் பெண்ணா?” என்றார் பேகம்அக்பர். தான் கேலியாகக் கேட்டதில் கோபமடைந்து விட்டார் என்று உணர்ந்த அவனுடைய கண்கள் பணித்தன.

6. Conclusion and Future Scope:

Abstractive Text Summarization is still in its infancy in the Indian language when compared to languages like English, French, Chinese, Arabic, German, and Spanish.

For Tamil language only extractive text summarizations techniques are available.Till now there are no research works done on Tamil Abstractive Text Summarization. In this article, we have proposed an Abstractive Text Summarization for Tamil using the m-T5 model.Model was trained on our inhouse dataset Tamil Kid's Stories which we have created.Proposed system was evaluated using Rouge Metrics such as Rouge1,Rouge2 and Rouge L.The challenges of putting the techniques forth and their solutions were investigated and evaluated.Our insight for future work is to test our model on large documents such as tamil literature books.Further we plan to train our model by increasing more data on our dataset to get better performance and results.

Reference

1. Joshi, A., Fidalgo, E., Alegre, E., & Alaiz-Rodriguez, R.: RankSum—An unsupervised extractive text summarization based on rank fusion. *Expert Systems with Applications*, 200, 116846 (2022)
2. Zhao, J., Yang, L., & Cai, X.: HetTreeSum: A Heterogeneous Tree Structure-based Extractive Summarization Model for Scientific Papers. *Expert Systems with Applications*, 210, 118335 (2022)
3. Du, Y., Li, Q., Wang, L., & He, Y.: Biomedical-domain pre-trained language model for extractive summarization. *Knowledge-Based Systems*, 199, 105964 (2020)
4. Verma, P., Verma, A., & Pal, S.: An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms. *Applied Soft Computing*, 120, 108670(2022)
5. Mohamed, M., &Oussalah, M.: An iterative graph-based generic single and multi document summarization approach using semantic role labeling and wikipedia concepts. In 2016 IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService) pp. 117-120. IEEE,(2016)
6. Rush, A. M., Chopra, S., & Weston, J. :A neural attention model for abstractive sentence summarization. arXiv preprint aXiv:1509.00685(2015)
7. Suleiman, D., &Awajan, A.: Multilayer encoder and single-layer decoder for abstractive Arabic text summarization. *Knowledge-Based Systems*, 237, 107791(2022)
8. Baykara, B., &Güngör, T.: Abstractive text summarization and new large-scale datasets for agglutinative languages Turkish and Hungarian. *Language Resources and Evaluation*, 1-35(2022)

9. Cai, X., Shi, K., Jiang, Y., Yang, L., & Liu, S.: HITS-based attentional neural model for Abstractive Summarization. *Knowledge-Based Systems*, 222, 106996(2021)
10. Cai, X., Liu, S., Yang, L., Lu, Y., Zhao, J., Shen, D., & Liu, T. COVIDSum:A linguistically enriched SciBERT-based summarization model for COVID-19 scientific papers. *Journal of Biomedical Informatics*, 127, 103999(2022)
11. Mohamed, S. S., & Hariharan, S.: A summarizer for Tamil language using centroid approach. *International Journal of Information Retrieval Research (IJIRR)*, 6(1), 1-15(2016)
12. Banu, M., Karthika, C., Sudarmani, P., & Geetha, T. V.: Tamil document summarization using semantic graph method. In *International conference on computational intelligence and multimedia applications (ICCIMA 2007)* ,Vol. 2, pp. 128-134. IEEE(2007)
13. Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., ... & Raffel, C.: mT5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934* (2020)
14. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J.: Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140), 1-67 (2020)
15. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems*, 30 (2017)

Sentiment Component Extraction from Dependency Parse for Hindi

¹Remya Panicker, ²Ramchandra Bhavsar, ³B. V. Pawar

(¹remya.panicker@gmail.com, ²rpbhavsar@nmu.ac.in, ³bvpawar@nmu.ac.in)

^{1,2,3}*School of Computer Sciences*

KBC North Maharashtra University

Jalgaon, Maharashtra (India)

Abstract. With the advent of Web 2.0, Natural Language Processing (NLP) gained attention and a new dimension in the NLP application arena. Sentiment Analysis is one such popular modern NLP application that tries to extract the feeling, emotions, opinions, etc. expressed in digital text using NLP techniques. Sentiment Analysis has become a subtle application over the last decade and found a firm foundation in AI applications. This is especially applicable to product services, reviews, and recommendations domains. It tries to quantify the sentiment better and helps understand the views expressed in a text leading to better decision-making.

There is a variety of approaches available for carrying out Sentiment Analysis ranging from naive to sophisticated machine learning approaches. However, less attention is being paid to linguistics aspects. We undertook a study to project sentiment analysis concerning linguistic dimensions of natural language text which is the least explored approach to Sentiment Analysis. Sentiment Components of a text are deeply rooted in its syntactic constituents. The only way to figure out the syntactic constituents is parsing. We have tried to portray the use of dependency parsing for extracting the sentiment components from Hindi input text. We have explored the applications of various dependency relations to derive the possible sentiment compositionality from Hindi Sentences. Our study and findings related to sentiment components extraction from dependency parse are presented in this paper with a Linguistic insight

Keywords: Sentiment Analysis, Sentiment Components, Dependency Parsing, Natural Language Processing (NLP)

1 Introduction

With the emergence of the participative web i.e. Web 2.0; there was a deluge of user-generated content on the internet. Social media flourished with user-generated digital content like a post, tweets, images, videos, blogs, etc. With this, there was an urge to extract the sentiments hidden inside this huge data. Multiple areas like Business, Marketing, Political, Social, etc. motivated to perform Sentiment Analysis on the digital text to extract the feelings, opinions, and sentiments expressed in the digital text. Sentiment Analysis finds its application in product review analysis on various shopping sites, recommendation systems, virtual chat-bots, political campaigning, natural language understanding, etc.

Sentiment Analysis [13][3] is the mechanism of classifying a piece of text according to the sentiment polarity (viz. negative, positive, neutral). The text can be a phrase, sentence, or document which contains some opinions, feelings, sentiments, etc. There exist various approaches to Sentiment Classification among them Lexicon Based Approach and Learning Based Approach are widely used. These approaches rely on a bag of word mechanisms, Part of Speech tagging, Feature Extraction, Dictionaries, Corpus, etc. The existing approaches mainly focus on polarity identification, but identifying the extent of emotions, feelings, comparisons, opinions, the strength of sentiment, etc. is still a challenge. This challenge can be overcome by accurately identifying the sentiment compositions that exist among the different words of a text. Present approaches least care about the sentiment compositions; they rather focus on only marking text as positive or negative. Sentiment Analysis is performed at two scales Fine-Grained and Coarse-Grained. Fine-grain sentiment analysis aims extraction of micro-level sentiment compositions from each word of a sentence. Coarse-Grained extracts sentiment on document level or sentence level.

For accurate sentiment analysis, it is necessary to identify the various sentiment components present in any text that contributes to the sentiment orientation. Components of sentiment may be opinions or sentiments expressed in the sentence, the object or entity about which some sentiment is expressed, the feature or attribute of any object, the person who is expressing the sentiment, feelings expressed, emotions, etc. It is difficult to identify these components while performing traditional sentiment analysis

which just relies on a bag of word mechanisms.

The NLP pipeline activities viz.

Segmentation, Tokenization, Part Of Speech (POS) tagging, Lemmatization, and Parsing [2], can be modified accordingly to extract sentiment composition. In this pipeline, POS taggers can be used to identify the underlying part of speech of the word. These POS categories can be employed to identify the sentiment components, e.g. adjectives denote the sentiment; Noun denotes the Holder of sentiment or an entity about which sentiments are expressed. These methods lack when there is a need to figure out the sentiment relationships that exist among the words of a sentence. Parsing is an important pipeline activity that involves the process of breaking the sentence into various parts of speech (POS) and understanding the meaning and relationship among the various POS components. Parsing is a process of determining whether a sentence is accepted by a given grammar or not. It is an important stage in the pipeline of NLP activities. The output of parsing is used for information extraction, word sense disambiguation, etc.

We bring to light that parsing can also help in understanding the sense, opinion, and sentiment expressed in the sentence. The output of the parser is a sentence diagram or parse representation which generally is a tree. Dependency Parsing [7] is a parsing technique that can perform syntactic and semantic analysis. In Dependency parsing, a syntactic structure of a sentence is described in terms of words with an associated set of binary relations between words of the sentence known as dependency relations.

In this paper, we have put forth a novel approach to performing sentiment analysis by identifying sentiment components and establishing the relationship between these components. The main contributions of the presented work are;

1. Concept of sentiment components.
2. Identifying the correlation between sentiment components at the syntactic and semantic parsing stages by analyzing the output of the dependency parser for the Hindi Language.
3. Identification of Sentiment Components and Formulation of Sentiment Relations.
4. Employing POS category ‘Verb’ for sentiment extraction i.e. deriving sentiments from action statements in Hindi.
5. The paper put forward an algorithm that takes the output of dependency parsers i.e. the dependency relations as input and identifies the various sentiment components and relations that exist in a sentiment expression for Hindi.

The paper is organized into 6 sections, section 1 is the introduction, section 2, describes how parsing contributes to sentiment analysis and related work, section 3, gives a brief description of various sentiment components and sentiment relations, section 4 presents details of dependency relations that help us to derive a particular sentiment attribute and components, section 5 depicts the algorithms, empirical testing used to implement this approach, section 6 highlights the limitation and amendments that have to be done in dependency parsing for in-depth sentiment analysis.

2 Related Work

Di Caro, Luig, Grella Matteo [4] proposed a method that performs sentiment analysis based on propagation rules that work at syntactic level. They used a Stanford Dependency Parser [12] for the English language and Tule [9] for the Italian language. It defines a syntactic based propagation rules that transfer sentiment values of the words to each other in the dependency parse tree.

R. Socher et al. [14] introduced a sentiment treebank that expresses the phrase or sentence into a fully labeled parse tree. They used Recursive Neural Tensor Network to address the problem of sentiment compositionality.

Kotov Artemy , Zinina Anna & Filatov Alexander [8] developed a semantic parser that constructed syntactic trees and a semantic representation-frame(set of valencies with semantic markers). This representation is used to identify the sentiment in the sentence by a computer agent. This method is devised only for the Russian Language

Agarwal Basant et al. [1] employed a concept extraction algorithm based on a concept parser. They used a ConceptNet ontology which provides the semantic of the text. This method had efficiently implemented common sense reasoning using common sense ontology and reasoning algorithms.

Li Dong et al. [10], presented a statistical parsing method for analyzing the sentiment structure of sentences directly. Here a sentiment grammar is built like Context-Free Grammar.

Liwen Zhang, Kewei Tu, and Yue Zhang [11], presented a neural network-based model for sentiment analysis by using Deep Sentiment Representation, Gaussian Mixture Vectors, Sentiment Grammar Models, and EMLo Embedding.

3 Sentiment Components and Sentiment Relations

Sentiment Components: S Sentiment Components are the words or groups of words that contribute to the sentiment structure formation of sentence/phrase/text.

In-depth sentiment analysis means performing a granular analysis of every word affecting the polarity of a sentence or phrase. It can be performed at the sentence or phrase level. Sentiment Components Extraction involves, a) extracting the entities i.e. the object/thing/subject for which sentiments are expressed, b) identifying the aspects/features/properties of the entities, c) determining the sentiment relationship that exists between an aspect and its entity, and d) identifying the holders, and targets of a sentiment. This provides a broader insight into sentiment orientations in a text. It gives the relationship that exists among the words of a sentence or phrase. For instance, the most important application of sentiment component extraction is that it mines the opinion, i.e. to identify what is being told about a particular entity, who is the holder of the sentiment, what is the polarity of the sentiment, etc.

A similar concept was put forth by Bing Liu,[3]. He has represented, opinions as a quintuple (entity, aspect, sentiment, holder, time). It is represented as (ei, aij, sijkl, hk, tl), where ei : name of the entity, aij : an aspect of ei , sijkl : sentiment expression on aspect aij of an entity ei, hk : is the holder of opinion, tl: Time when the sentiment was expressed. The important task of fine-grained sentiment analysis is to identify all five components of the quintuples. This paper focuses on identifying those quintuples elements along with some other components. Following are the important remark about all five components of quintuples by considering an Example:

E.g. 1: रमेश को सोनी मोबाइल का कैमरा उत्तम लगा।

Transliteration: Ramesh ko sonī mobāil kā kaimarā uttam lagā

Translation: Ramesh liked the Sony mobile camera

The quintuple will be (सोनी मोबाइल, कैमरा, उत्तम, रमेश, Present time)

There are various types of sentiment expressions other than the above. Sentiment analysis can be classified as Opinion Analysis, Intent Analysis, Emotion Analysis, etc. which constitutes a Sentiment Taxonomy [17], which includes various types of sentiment expression. Each type of sentiment expression is composed of various types of sentiment components. We will discuss only the Feedback and Opinion Analysis in this paper. The scope of the paper considers components like entity, aspect, sentiment, time, holder, etc. Other than the above-mentioned components many other

components contribute to the overall sentiment analysis viz. Emotions (Joy, anger, love, care, etc.), reaction to an action, sarcasm, comparison, negations, intensifications, point of view, etc. Following are some of the important sentiment components identified for Feedback/ Opinion Analysis:

Entity: Entity broadly refers to an object, person, place, topic, issue, event, etc. about which something is expressed in the text. It is similar to Name Entity Recognition (NER) in NLP. In E.g. 1 सोनी मोबाइल्स is the entity. It is usually Noun or Noun Phrase. It is also referred to as objects.

Aspect: Aspect denotes the attribute of an entity. They are also referred to as features, attributes, facets, etc. It implies the characteristics of an entity. It is usually a noun and a noun phrase, but it can be a verb, or adjective too. In above example 'कैमरा' is the aspect of entity 'सोनी मोबाइल्स'

Qualifier: It's the feature of the aspect. It is the characteristics of any aspect of an entity.

Sentiment: The sentiment is the opinionated word that denotes the polarity (positive, negative, or neutral). It is usually an adjective or an adverb, sometimes verb too denotes sentiment. In the E.g. 1 'उत्तम' is the sentiment denoting the word.

Holder: Holders of sentiment are the person who is expressing the sentiment or conveying the opinion. It becomes vital to justify a sentiment from the perspective of a holder, so identifying a holder is an important task. In the above ex. 'रमेश' is the holder of the sentiment expressed.

Time: Time denotes the time (date) or tense when the opinion or sentiment was expressed. Sometimes tense and time is used to justify the sentiment and opinion so time extraction becomes important.

Intensifier: Intensifiers are responsible for the escalation of the extent of sentiments in a word. For E.g. 2. रमेश बहुत अच्छा विद्यार्थी है पर खराब खिलाड़ी है. Transliteration: Ramesh bahut achchhā vidyārthī hai par khrāb khilādī hai. Translation: Ramesh is a very good student but a bad player). Here 'बहुत' is increasing the intensity of the sentiment word 'अच्छा' which converts a positive statement into a highly positive one.

Sometimes action denoting sentences too denotes sentiments. POS category Verb too affects the overall sentiment composition of a sentence. Most of the Sentiment Analysis approaches just rely on adjective and adverbial words; they ignore the significance of verbs or action-denoting words while extracting the sentiments. We have identified some sentiment components that are related to action-denoting statements some of them are discussed below.

Consider an example:

E.g.3. डाकू ने घर जलाया।

Transliteration: Dākū ne ghar jalāyā

Translation: The robber burnt the house.

Action: The verb in the sentence which denotes some action/activity is an action component. In E.g. 3 ‘जलाया’ is the action components. Which conveys a negative sentiment.

Action Subject/Agent: The doer/initiator of an action is the action subject. In E.g. 3 ‘डाकू’ is the doer or responsible for action 2 ‘जलाया’.

Action Object/Patient: Action Object represents the recipient of the action. It is the entity which is affected by the action. In E.g. 3 ‘घर’ is the entity affected by the action ‘जलाया’.

Sentiment Relation: The relationship that exists between sentiment components is referred to as Sentiment Relation. It's the same like dependency relation but they denote the sentiment relationship that contributes to the sentiment extraction and helps in understanding the sentiment orientations among the words. We have identified some of the common relationships that exist in opinioned statements which are discussed below:

Entity and Root : Root Entity about which sentiment/feedback/Opinion is expressed

Attribute and Entity: Feature/Property/Attribute of Entity

Aspect and Attribute: Feature/Aspect/Property of Attribute.

Aspect and Qualifier/ Aspect and Sentiment: Quality/Sentiment of Entity, Attribute or Aspect.

Qualifier and its Descriptor: Descriptor of Qualifier

Qualifier and its Holder: Holder or Person expressing opinion or feedback

Qualifier and Time: Time at which the opinion or feedback was expressed

Following are the relations for action denoting statements:

Subject of Action: It is the action doer/initiator

Object of Action: Recipient of the action.

4

Employing Parsers for Sentiment Component Extraction

In Natural Language Processing (NLP) parsing involves the process of breaking the sentence into various parts of speech and understanding the meaning and relationship among the various POS components. It is the process of determining, if a sentence is accepted by a given grammar. It is also called Syntactic Analysis. Syntactic analysis is an important stage in the pipeline of NLP activities. The output of parsing is used for information extraction, word sense disambiguation, etc. It also helps in understanding the sense and sentiment expressed in the sentence. The outputs of the parser are a sentence diagram or parse representation which generally is a tree. Parsing requires a grammar to produce parse trees or graph. There are various techniques of Parsing:-*Deep Parsing*: This technique of parsing aims to create the complete syntactic structure of a sentence. It is used in complex NLP applications. It is also known as Full Parsing. *Shallow Parsing/ Partial Parsing*: Many times it is not desired to have a complex and complete parse tree for all inputs, for this partial parse or shallow parse is sufficient. *Chunking*: It is similar to partial parsing; it identifies

and classifies the flat, non-overlapping segments of a sentence. *Dependency Parsing*: In, dependency parsing a syntactic structure of a sentence is described in terms of words with an associated set of binary relations between words of the sentence.

Sentiment analysis is mainly carried out by the bag of word approach. To identify the sentiment compositionality, it is necessary to understand the semantic organization of a sentence. Parsing can contribute to deriving the relationships that exist among the words and POS categories. Dependency Parsing is one such important method. Dependency Parsing [7] is a parsing technique in which the output of the parser is a directed graph. The directed graph has directed arcs from headword to dependent word showing the relations that exist between the words. These relations give a better understanding of the various words that makes a sentence. The major advantage of using a dependency parser for Hindi is the ability of Dependency parsers to efficiently deal with the free word order nature of the language. Some of the work reported that performs sentiment analysis at the parsing stage is discussed in the following section.

5 Sentiment Analysis based on Dependency Parsing

The output of a Dependency Parser [7] is a directed graph with directed arcs from head to dependent. It is also called typed dependencies. Dependency Relations are binary grammatical relations that exist between two words. The syntax of writing the dependency relation is; the relation name (head, dependent). Dependencies are pairs of the head (known as governor or regent) and dependent. The advantage of using a dependency parser for Hindi is its ability to deal with free word order languages.

Consider the following example:

E.g. 4.: जवान ने दुश्मन को मारा।

Transliteration: Javān ne dushman ko mārā

Translation: Soldier kills enemy

The POS (Part of Speech) categories for each word in above example are:

- (जवान, NOUN),
- (ने, Adposition),
- (दुश्मन, NOUN),
- (को, Adposition),
- (मारा, VERB).

The dependency relations that occur in above example are:

- Root Word: root(root, मारा)
- Nominal Subject: nsubj('मारा', 'जवान')
- Prepositions/Postpositions: case('जवान', 'ने'), case('दुश्मन', 'को')
- Object: obj('मारा', 'दुश्मन')

From the dependency relation, we can figure out the subject, object, and action.

We performed a deep study of these dependency relations. This study helped us to identify the dependency relations that contributed to sentiment component extractions. From this study, we figured out the combinations of dependency-relations, and POS categories(of Head and Dependent) which helped us to extract the sentiment components and their relationship. From this, we framed multiple rules which contributed to in-depth sentiment analysis.

The Universal Dependency (UD) Project undertaken by Joakim Nivre et.al [5] developed a cross-lingual Universal Dependency Tree Bank for more than 50 languages. This Universal Dependency Project consists of data sources and performs segmentation, tokenization, POS tagging, lemmatization, and dependency parsing. We used the dependency relation from the UD project to analyze the sentiment component.

The following section describes the various Dependency Relation [16] that help in extracting the various sentiment components like entity, aspect, holder, sentiment and time from a Hindi sentence. Table. 1 shows some of the relations and its sentiment derivations.

Table 1. Sentiment Components Extracted from Dependency/Syntactic Relation

Syntactic Relation	POS of Head	POS of Dependent	Sentiment Component denoted by head	Sentiment Component denoted by dependent
nsubj:Nominal Subject	Verb	Noun/Pronoun	Action Sentiment	Action Subject / Doer / Agent
	Adjective	Noun/Pronoun	Sentiment/opinion	Entity / Aspect
	Noun	Noun/Pronoun	Aspect	Entity/Object
obj:Object	Verb	Noun/Pronoun	Action Sentiment	Action Object/ Receiver/ Patient
iobj:Indirect Object	Verb	Noun/Pronoun	Action Sentiment	Action Addressee
xcomp: open clausal complement	Verb	Adjective	Action Sentiment	Feature of Sentiment (Sentiment Modifier)
obl:oblique nominal	Verb	Noun/Pronoun	Aspect	Entity
nmod:Nominal Modifier	Noun	Noun/Pronoun	Aspect	Entity
	Adjective	Noun/Pronoun	Sentiment/Opi- on	Aspect/Entity
advcl: Adverbial clause modifier	Verb	Verb	Sentiment Reaction	Sentiment Action

6 Experiment and Empirical testing

Approach: For the experimental setup, we have implemented a dependency parser from the Universal Dependency (UD) Project (Joakim Nivre et.al[5]) for the Hindi language. Used ufal.udpipe package to implement the model. The output of this is dependency parse relation. This relation is given input to the ‘Sentiment Component Extraction Algorithm’, which after execution classifies the words from the input text into their corresponding sentiment components. The output of the algorithm is the Sentiment Component List (Related Sentiment Components) and Sentiment Component Relation List (Sentiment Relationships among words).

Sentiment Component Extraction Algorithm: A rule-based algorithm is devised to identify the type of each word (POS Category). It helps in the identification of sentiment components by using the following aspects:

- Type of Dependency Relation.
- Head and Dependent words from dependency relation
- POS categories of Head and Dependent words

Data Structures used in Algorithm:

- QUALIFIERS: List of sentiment denoting words from given text.
- ASPECT: List of aspect denoting words from given text.
- AGENT: List of doer/agent of an action
- ACTION: List of action denoting words from given text.
- ENTITY: List of entity denoting words from given text.
- HOLDER: List of words denoting holder of sentiment from a given text.
- INTENSIFIER: List of intensifiers from the given text.
- OBJECT: List of receiver/patient of some action/verb from a given text.
- SENTIMENT COMPONENT LIST: Related Sentiment Components
- SENTIMENT COMPONENT RELATION'S LIST: Sentiment Relationships among words

Sentiment Component Extraction Rules: Using combinations shown in Table. 1 the Sentiment Component Extraction Rules are crafted to figure out the sentiment compositions. It is not limited to the combinations mentioned in the table, but there is a scope for the addition of newly identified rules.

Example of Sentiment Component Extraction Rules:

IF Relation equals NSUBJ Then

IF HEAD's POS category is ADJECTIVE Then

```

Add Dependent word in ‘QUALIFIER’ list.
Add Head word in ‘ASPECT’ list
Create a sentiment relation ENTITY_QUALIFIER (Head word, Dependent word)
ELSE IF HEAD’s POS category is VERB Then
    Add Dependent word in AGENT list
    Add Head word in ACTION list
    Create a sentiment relation ACTION_AGENT (Head word, Dependent word)
ELSE
    Add Dependent word in ‘ENTITY’ list.
    Add Head word in ‘ASPECT’ list
    Create a sentiment relation ENTITY_ASPECT (Head word, Dependent word)
END IF

```

Algorithm:

Input: Dependency relations (From Dependency Parsers)
Output: *Sentiment Component List: Related Sentiment Components, Sentiment Component Relation’s List: Sentiment Relationships among words*

Step 1: Read the relations from dependency parser.
Step 2: Identify the Relation type and POS Category of Head and Dependent words.
Step 3: Apply the “Sentiment Component Extraction Rules”
Step 4: Populate the respective “Sentiment Component Lists”
Step 5: Populate the respective “Sentiment Component Relation’s List”
Step 7: Print the relation.
Step 8: End

Experimental Setup: Dataset of 100 manually annotated simple Hindi Sentences (Sentiment Denoting Sentences) were created for evaluating this approach. Dependency parser from the Universal Dependency (UD) Project (Joakim Nivre et.al[5]) for the Hindi language(ufal.udpipe package) was used to derive dependency relation pairs. These relations were provided to ‘Sentiment Component Extraction Algorithm’.

The output of the algorithm is the Sentiment Component List (Related Sentiment Components) and Sentiment Component Relation List (Sentiment Relationships among words). Following sample example denotes this process:

Sample:

Sentence: "रमेश बहुत अच्छा विद्यार्थी है पर खराब खिलाड़ी है"

Transliteration: Ramesh bahut achchhā vidyārthī hai par kharāb khilādī hai

Translation: Ramesh is a very good student but a bad player.

The above sentence is fed to the parsing module, which gives the set of following relations:

- [((None, 'TOP'), 'root', ('विद्यार्थी', 'NOUN')),
- ((विद्यार्थी, 'NOUN'), 'nsubj', ('रमेश', 'PROPN')),
- ((विद्यार्थी, 'NOUN'), 'amod', ('अच्छा', 'ADJ')),
- (('अच्छा', 'ADJ'), 'advmod', ('बहुत', 'ADV')),
- ((विद्यार्थी, 'NOUN'), 'cop', ('है', 'AUX')),
- ((विद्यार्थी, 'NOUN'), 'conj', ('खिलाड़ी', 'NOUN')),
- ((खिलाड़ी, 'NOUN'), 'cc', ('पर', 'CCONJ')),
- ((खिलाड़ी, 'NOUN'), 'amod', ('खराब', 'ADJ')),
- ((खिलाड़ी, 'NOUN'), 'cop', ('है', 'AUX'))]

These relations comprise the words, its POS category and name of the syntactic relation that exists between the two words. These relations are given as an input to the algorithm. The algorithm produces following output.

Output of Sample: The output is a sentiment relation pair. Its interpretation is described below:

- *ENTITY_ASPECT*(विद्यार्थी, रमेश): विद्यार्थी is Aspect of Entity रमेश
- *ASPECT_QUALIFIER*(अच्छा, विद्यार्थी): अच्छा is qualifier/sentiment of/by Aspect विद्यार्थी
- *ASPECT_QUALIFIER*(खराब, खिलाड़ी): खराब is qualifier/sentiment of/by Aspect खिलाड़ी
- *QUALIFIER_INTENSIFIER*(अच्छा, बहुत): अच्छा is intensified/described/ बहुत →

Evaluation:

Similar to above sample, some 100 sentences were fed to the model. The output was evaluated using he metrics discussed below. The common evaluation metrics evaluates the conceptual "distance" between the candidate parse generated by the parser, and the correctly annotated solution (the "gold standard"). For evaluation we are using Gold Standard, here sentences are annotated by hand. Table 2. Denotes the Exact Match (EM) metrics which is 70% i.e. about 70 sentences were correctly labeled by the model.

About 380 relations (identified manually) were there in 100 sentences. Labeled attachment Score (LAS) Refers to the proper assignment of a word to its sentiment components along with the correct sentiment relation. Table. 3 denotes the LAS score, out of 380 relations 330 relations were correctly identified.

Table. 4 denotes the Precision, Recall and F-Score for total sentences, considering average of total labeled sentences. Table. 5. denotes metrics for each relation, considering average of total labeled Sentiment Relations

Exact Match (EM) Metrics:

Table 2. EM Metrics

Total Number of Sentence Correctly Labeled	Total Number of Sentence	Accuracy
70	100	70%

Table 3. Labeled attachment Score

Total Sentiment Relations	Correctly Labeled Relations	LAS
380	330	0.86

Precision, Recall and F-Score:

For Total Sentences: Considering average of total labeled sentences

Table 4. Precision, Recall and F-Score for Total Sentences

Accuracy	Precision	Recall	F-Score
0.88	0.87	0.76	0.80

For Each Relation: Considering average of total labeled Sentiment Relations

Table 5. Precision, Recall and F-Score for Sentiment Relations

Sentiment Relation	Accuracy	Precision	Recall	F-Score
Entity Root	0.80	0.75	0.60	0.67
Attribute Entity	0.70	0.86	0.60	0.71

Aspect Attribute	0.83	0.80	0.70	0.75
Aspect Qualifier	0.80	0.88	0.70	0.78
Qualifier Descriptor	1	1	1	1

7 Limitations of This Approach

From the study of the various work undertaken in the field of syntactic, semantic, and statistical parsing, it is observed that the output of a parser only deals with the deriving relationship between the grammatical structures of the sentence. Once the parse tree or parse representation is generated then the sentiment analysis task is performed. This restricts us from performing in-depth sentiment analysis of a text. The above study helps us to derive only the sentiment that is explicitly mentioned in the text i.e. terms reflecting sentiment are present in the sentence. It works efficiently with simple sentences. It may fail in many instances where sentiment compositionality is complex. There may exist sentences that cannot be processed merely by a dependency parser.

An Implicit sentiment denoting sentences is difficult to process with a dependency parser. In these types of sentences, the words do not reflect any sentiment directly.

E.g. पिक्चर देकते ही मुझे नींद आ गयी। (Pikchar dekate hī muze nīanda ā gayī/ I fell asleep as soon as I saw the picture)

A Sarcastic Sentence; is a sentence in which words are used to express the opposite of what you want to say, especially to insult someone, to show irritation, or just to be funny. E.g. क्या बढ़िया कार है, दो दिनों में ही बंद हो गयी! (Kyā badhiyā kār hai, do dinoan mean hī banda ho gayī/ what a great car it was stopped in two days).

Comparative Sentence; is a type of sentence in which a comparison is made between two objects and entities. E. g. :सोनी का मोबाइल सॉमसंग से अधिक अच्छा है। (Sonī kā mobāil saōmasanga se adhik achhā hai/ Sony's mobile is better than Samsung)

Feeling, Emotion; Emotion Based Expressions are difficult to analyze. Various emotions like anger, joy, fear, anxiety, love, jealousy, etc. cannot be fully covered by dependency parsing. Ex. पिता अपनी बेटी की करतूत पर हैरान है और उसे शाप देते हैं। (Pitā apanī betī kī karatūt par hairān hai aur use shāp dete haian /Father is shocked at his daughter's handiwork and curses her)shows anger but there's no way to classify it as negative.

8 Future Scope and Conclusion

The dependency parsers just depict the bigram dependencies. The output of dependency parsers is insufficient to represent the overall sentiment relations among words in a text. The Hindi Language sentiment expression consists of varied

relationships, and emotional expression, that has to be modeled into proper Sentiment Based Typed Dependencies. Hence there is a need for devising and formulating Sentiment Based Typed Dependencies. It is also important to decide the hierarchy of sentiment-based typed dependencies to incorporate all sentiment expressions. Designing styles of representation for typed relations using a suitable data structure is also crucial in sentiment analysis. Parsers dedicatedly to sentiment extraction can be designed to facilitate sentiment analysis and language understanding. Most of the cited work is done in English language or foreign languages. There is a scarcity of such approaches in regional languages like Hindi, Marathi, etc. So, there is a scope for such research in these languages. Also, such parsers can reduce the flaws and errors in currently used techniques by providing a correctly interpreted sentence as an output of the parsing phase

References

1. Agarwal B., Poria S., Mittal N., Gelbukh A., & Hussain A., 'Concept-Level Sentiment Analysis with Dependency-Based Semantic Parsing: A Novel Approach', Cognitive Computation issn: 1866-9956, pp 487–499
2. Aho A V, Sethi R, Ullman J D: "Compilers: Principles, Techniques and Tools", Second Edition, Pearson Addison Wesley(2007)
3. Bing Liu : "Sentiment Analysis and Opinion Mining", Morgan & Claypool Publishers, (May 2012).
4. Di Caro, Luig, Grella Matteo : "Sentiment analysis via dependency parsing", Computer Standards & Interfaces, vol.35,Issue. 5,C, pp 442–453(2013)
5. Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajíč, Christopher Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman : "Universal Dependencies v1: A multilingual treebank collection", In Pro of the 10th International Conference on Language Resources and Evaluation (LREC 2016),pp 1659–1666, (2016)
6. Jurafsky Daniel & Martin James : "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Second Impression, Pearson Education Inc.,(2009).
7. K'ubler S, McDonald R, Nivre J., "Dependency Parsing" : Synthesis Lectures Human Language Tech, vol. 1, pp. 1–127, (2009).
8. Kotov Artemy , Zinina Anna & Filatov Alexander : "Semantic Parser for Sentiment Analysis and the Emotional Computer Agents", in Porc AINL-ISMW FRUCT (2015)
9. L. Lesmo, "The Turin university parser", In proc : Proceedings of EVALITA vol. 9, (2009)
10. Li Dong, Furu Wei, Shujie Liu, Ming Zhou, Ke Xu : "A Statistical Parsing Framework for Sentiment Classification", Computational Linguistics,vol.41, issue. 2,pp 293-336,June (2015)
11. Liwen Zhang, Kewei Tu, and Yue Zhang : "Latent Variable Sentiment Grammar". In the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019), Florence, Italy, July 28 – August 2,(2019).
12. M.C. de Marneffe, B. MacCartney, and C. D. Manning : "Generating Typed Dependency Parses from Phrase Structure Trees," in proc LREC,vol. 6,(2006).
13. Pang Bo and Lillian Lee : "Opinion mining and sentiment analysis", Foundations and Trends in Information Retrieval, vol. 2,issue. 1-2, pp 1–135, (January 2008)

14. Richard Socher, Perelygin Alex, Wu Jean, Chuang Jason , Manning Christopher , Ng Andrew, Potts, Christopher. : "Recursive deep models for semantic compositionality over a sentiment treebank", In proc EMNLP 2013, 1631-1642(2013).
15. Straka Milan & Strakova Jana : "Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe", In Proc of CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies vol. 1, pp. 1–127,(2017)
16. Universal Dependency, "<https://universaldependencies.org/u/dep0/>", accessed on 14/11/2022.
17. R. Panikar, R. Bhavsar and B. V. Pawar: "Sentiment Analysis: a Cognitive Perspective," 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022, pp. 1258-1262, doi: 10.1109/ICACCS54159.2022.9785027 (2022)

Change Detection Algorithm for Vegetation Mapping using Multispectral Image Processing

Neelam B V D Soujitha¹, Mohammad Neelofar Jaha¹, Mahali Tirumala Raju¹,
Kakamanu Christy Victor¹ and Radhesyam Vaddi²

Student, Department of IT, VR Siddhartha Engineering College,
JNTU-Kakinada, Vijayawada, India.

Assistant Professor, Department of IT, VR Siddhartha Engineering College,
JNTU-Kakinada, Vijayawada, India.

¹durgasoujitha234@gmail.com, ¹neelofarjaha191202@gmail.com ,
¹tirumalarajumahali@gmail.com,
¹kakumanuchristyvictor@gmail.com, ²syam.radhe@gmail.com

Abstract. Agriculture is an important sector of the world. Agriculture's contribution to the nation's prosperity cannot be overlooked, despite the industry's significant role in the global economy. Agriculture in any geographic area is constantly changing. Changes are detected to build a knowledge base in order to ensure the preservation of these agricultural characteristics in particular location. The process of identifying differences in two different places or the state of an object or phenomenon by observing it at different times is known as change detection. This method is typically used to change the earth's surface two or more times. Here the primary source of data is geographic which is taken from Google Earth Engine(GEE) and is typically in digital (e.g., satellite imagery) can be used. The history of change detection begins with the history of remote sensing

Keywords: Change detection, quantum geographic information system, google earth engine, remote sensing images, google colab, multispectral Images

1 Introduction

In recent decades, remote sensing data have been employed widely as primary resource for change detection. Understanding the connections and interactions between human and natural events is crucial for better decision-making, and for this accurate change detection of Earth's surface features is necessary. The method of finding differences between scenes of the same location taken at several periods is known as remote sensing change detection (RSCD). Change detection is a method that assesses how a certain area's characteristics have evolved over the course of two or more time periods. Aerial or satellite images of the area obtained at various dates are frequently compared in order to discover changes. Recognizing the type and location of changes, quantifying changes, and evaluating the accuracy of change

detection results are the general goals of change detection in remote sensing. The change detection techniques are divided into seven categories for ease of reference: (1) algebra; (2) transformation; (3) classification; (4) advanced models; (5) approaches using Geographical Information Systems (GIS); (6) visual analysis; and (7) other approaches. Absolute and relative change detection are the two types of change detection. Absolute change detection draws attention to the specific changes, such as the transition from forest to grassland. Although something has changed, relative change detection does not identify what that change is.

Utilizing several temporal data sets, change detection entails quantifying temporal impacts. Satellite data is frequently employed when someone is interested in tracking changes over vast areas and at frequent intervals. Executing checks against two states—the new state and the current state—is the main change detection technique. One of these states must be rendered if it diverges from the other, indicating that something has changed. Change detection is of two different forms. The first is supervised change detection, and the second is unsupervised change detection. The outcomes of the GIS study demonstrate the use of change detection information in crop insurance by assisting in the assessment of field damage. In order to conserve crops and reduce yield losses, farm management decisions can ultimately benefit from the GIS information gathered.

Due to its ability to quantitatively analyse the geographical dispersion of the target population, change detection is a crucial procedure in the management and monitoring of natural resources and urban growth. Assessment of deforestation, monitoring shifting cultivation, study of changes in vegetation phenology, seasonal variations in pasture production, damage assessment, and crop monitoring are just a few of the many areas where change detection is helpful.

1.1 Applications of Remote Sensing

In order to identify, measure, and analyse specific items, areas, or phenomena properties without coming into contact with them directly, remote sensing is the science and technology that enables this. Land use mapping, weather forecasting, environmental research, studying natural hazards, and resource exploration are some of the uses for remote sensing. The electromagnetic radiations that an object emits or reflects serve as the source of remote sensing data, which is subsequently used to identify and categorize the object. Its data can be used to track changes over time and receive the most recent land use patterns over broad areas at any one time. It can be used to update wetland delineation, asphalt conditions, and road maps. Regional planners and administrators utilise this data to frame policy decisions for the region's overall development.

2 Related Work

Land Cover Clustering for Change Detection using Deep Belief Network(2022) [1]:

Authors uses Clustering for unlabeled data, Deep brief network(deep learning model) for change detection. It complicates the entire clustering process, and as a result, the accuracy of change detection sometimes suffers.

Optical Satellite Image Change Detection Via Transformer-Based Siamese Network(2022) [2]:

Convolutional neural networks(CNN),Natural language processing(NLP), vision Transformer(ViT) are used. Siamese extensions of ViT networks that outperform in tests on two open change detection datasets. The effectiveness and superiority are demonstrated by experimental results on real datasets.

Using Hyperspectral Reconstruction for Multispectral Images Change Detection(2022) [3]:

The original multispectral data is used to generate hyperspectral data with richer spectral information, which is then used to detect changes. They uses reconstruction algorithms for image reconstruction. The results of the experiments show that the hyperspectral reconstruction method improves the accuracy of multispectral images.

A CNN-Transformer Network With Multiscale Context Aggregation for Fine-Grained Cropland Change Detection(2022) [4]:

Authors uses CNN-transformer network ,Multiscale context aggregation (MSCANet) as the feature extractor. MSCANet demonstrates its space and computation complexity advantages. All of the results fully demonstrated the MSCANet's capability in efficient and effective cropland Change detection.

Style Transformation-Based Spatial–Spectral Feature Learning for Unsupervised Change Detection(2022) [5]:

Style transformation-based change detection algorithm with spatial-spectral feature learning (STFL-CD),Detection Network With Attention Mechanism, and Convolutional neural networks(CNN) are used by authors. They tried to remove the influence of "same object with different spectra" and had some shortcomings. Further research will be conducted in nonlinear ST methods in the future to deal with the multitemporal HSI change detection task more effectively.

A Survey on Deep Learning-Based Change Detection from High-Resolution Remote Sensing Images(2022) [6]:

Deep learning-based change detection using high-resolution images. It discusses the most popular feature extraction deep neural networks as well as the mechanisms for building them. Describes the change detection framework at first. After that, the methods are classified based on the deep network architectures used.

Detection of Urban Built-Up Area Change From Sentinel-2 Images Using Multiband Temporal Texture and One-Class Random Forest(2021) [7]:

The proposed multitemporal image classification method based on one-class Random forest (iOCRF) focuses on the built-up area change. It is suitable for including multiband temporal texture, and is data dimensionality insensitive. More research is needed to assess the performance of the proposed method in other study areas and changes of interest.

Image based Land Cover Classification for Remote Sensing Applications(2021) [8]:

They deals with Convolution neural network (CNN) in deep learning, supervised and un-supervised classification methods. Here the accuracy of land cover classification is heavily dependent on the time and location of the image capture.

Land Use and Land Cover Classification using Recurrent Neural Networks with Shared Layered Architecture(2021) [9]:

Authors uses Recurrent Neural Network (RNN) and Shared Layer Recurrent neural networks(SLRNN). SLRNN outperforms others in terms of accuracy. Not all pixels considered by SLRNN were correctly classified, but some were incorrectly classified.

Deep Learning Approaches to Earth Observation Change Detection(2021) [10]:

Two distinct approaches to detecting change (semantic segmentation and classification). Both use convolutional neural networks to address these specific requirements. Further developed and applied in post-processing workflows for a wide range of applications

Recent Applications of Landsat 8/OLI and Sentinel-2/MSI for Land Use and Land Cover Mapping: A Systematic Review(2020) [11]:

Advising the scientific community on how to use L8/OLI and S2/MSI data to gain a thorough understanding of land use land cover (LULC) mapping. Change

detection in various landscapes, particularly agricultural and natural vegetation scenarios. When using representative samples, classification models tend to achieve higher accuracies.

A Land Cover Change Detection Method Combing Spectral Values and Class Probabilities(2020) [12]:

Here spectral-based direct comparison (SDC) method and class probability-based direct comparison (CPDC) methods are used. To obtain class probability information, the supervised change detection method requires manual selection of training samples.

Deep Learning for change detection in Remote Sensing Images (2020) [13]:

Learn complex features of remote sensing images automatically using a large number of hierarchical layers. Among all classifiers, the Random Forest Classifier has the highest accuracy. The Artificial Neural Network (ANN) predicted soil fertility, crop yield, and crop yield with the highest accuracy.

Deep Learning for Land Cover Change Detection(2020) [14]:

The presented Long short term memory (LSTM) approaches are adaptable to a variable number of image sequences. The chosen pre-processing improves the water classification while avoiding effectively reducing the dataset.

A Deep Learning Architecture for Visual Change Detection(2018) [15]:

This paper proposes a parallel deep convolutional neural network (CNN) architecture for localising and identifying differences between image pairs. Change-Net is a deep architecture for detecting differences between pairs of images. Change-Net outperforms the competition.

3 Proposed Work

3.1 Datasets

We have used Google Earth Engine (GEE) for taking the input satellite images. Google Earth Engine enables users to visualize and analyze satellite images of our planet. It is a platform for geospatial analysis in the cloud that allows users to see and examine satellite photos of our world.



Fig. 1. Images of Switzerland from Google Earth Engine(2012)



Fig. 2. Images of Switzerland from Google Earth Engine(2015)

3.2 System Architecture

Deep Active Learning is supported in the implementation of the Siamese U-Net model with the pre-trained ResNet34 architecture as an encoder for the Change Detection task on the Remote Sensing dataset.

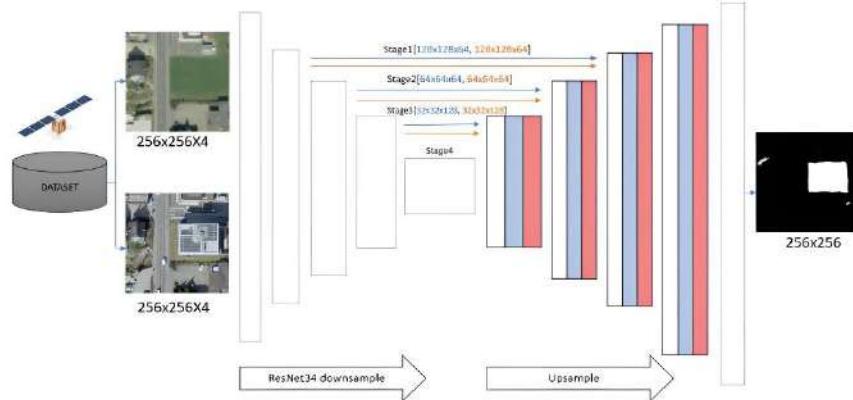


Fig. 3. Architecture diagram of Siamese Neural Network

A Siamese neural network uses ResNet34 as an encoder. The images are cropped to the correct size. The multispectral image is sent into the convolutional layers of the neural network to produce a feature vector. A feature vector holds all of the features of a picture. Time stamp T1's image X1 generates the feature vector $f(X_1)$ that corresponds to image X1. The picture X2 of timestamp T2 similarly generates the feature vector $f(X_2)$ corresponding to image X2. The identical Convolution layers receive both X1 and X2 simultaneously. The sent Convolution layers have the same parameters. The feature vector is a 128-bit encoded vector. Our Siamese neural network's output is a difference vector of both the feature vectors.

4 Algorithm and its Description:

A sort of artificial neural network is the siamese neural network that employs the same weights to compute equivalent output vectors from two separate input vectors while operating in parallel. A precomputed version of one of the output vectors frequently serves as a benchmark for comparison with the other output vector. These networks compare the feature vectors of the inputs to determine how similar they are. We conduct verification, identification, or recognition tasks using Siamese networks; the most well-known applications are face recognition and signature verification.

A siamese network consists of two comparable neural networks that each take one of the two input images. The contrastive loss function uses the final layers from both networks to determine how similar the two images are. The Siamese architecture seeks to distinguish between input images rather than categorise them.

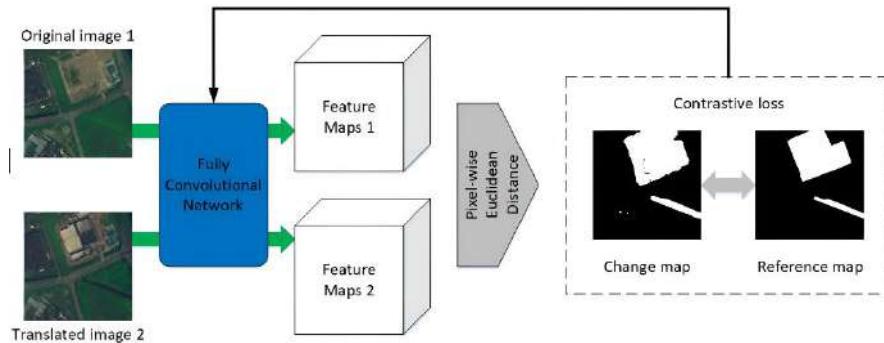


Fig. 4. Dataflow diagram of the Siamese detector, the changed pixels are white while the unchanged are black.

There are two sister networks that are identical neural networks with the same weights. One of these networks receives a feed of each image in the image pair. It optimises the networks using a contrastive loss function.

The contrastive loss function is given as follows:

$$(1-Y)1/2(D_w)^2 + (Y)1/2\{\max(0, m-D_w)\}^2 \quad (1)$$

where Dw is referred to as the Euclidean distance between the sibling Siamese networks' outputs.

Mathematically the Euclidean distance is:

$$(\{G_w(X_1) - G_w(X_2)\})^{1/2} \quad (2)$$

where Gw is one of the sister networks' outputs. The data pair for input is X1 and X2.

Algorithm: Siamese Neural Network

Input: High-Resolution satellite images

Output: Change map

1. Begin

2. Assume that Q = (cQ, tQ) and R = are the two input images (cR, tR)

3. Calculate the Euclidean distance between the matrix representations of the two matrices as follows: $D\phi(Q, R)=\|\{\phi(cQ) \oplus \phi(tQ)\} - \{\phi(cR) \oplus \phi(tR)\}\|_2$, to determine how semantically linked the two matrices are.

The caption representations for the query and the candidate matrix are (cQ) and (cR), respectively, and the matrix representations are (tQ) and (tR), respectively.

4. To ensure that D(Q, R) is modest, contrastive loss is specified as a quadratic function of table pair distances (close to zero)

5. If margin m is equal to or more than zero else and table R is comparable to table Q.

$$L(y, \phi, Q, R) = 1/2 (1 - y)D\phi^2(Q, R) + 1/2 y \{\max(0, m - D\phi(Q, R))\}^2 \quad (3)$$

6. The pair's genuine label is y.

7. Pairs with distance scores below this threshold are considered similar, whereas pairs with greater distance scores are judged dissimilar. Labels for a binary classification are obtained by thresholding the distance at half of the margin, $m/2$.

8. Output is the Images' Difference Label. $L(Q, R)$

As everything evolves through time, it can be quite beneficial to comprehend and measure this change. For instance, examining the infrastructural development of a city or town over time can be used to estimate its level of economic success. Every model has to be having the ability to recognise the changes in which we are interested while being able to ignore those that are not relevant to our use case. To detect changes in structures over time, it may be necessary to ignore changes in things like roads, trees, and water bodies. This is more of a data normalisation issue because images taken over time would have variations that would be difficult to constantly account for. When capturing facial photos with a smartphone, the lighting and orientation settings will never be the same. Photos acquired by satellites may be affected by changes in cloud cover, sunlight reflection, and the azimuthal and elevational angles of the satellite itself.

The simplest and most direct way to measure change, using a pixel difference between elements, is no longer an option. There are more clever ways to do this, but the results are still quite subpar. Two photos (taken at two separate timestamps) are fed into a Siamese neural network, which uses a series of convolutional layers to create higher-order feature vectors for each image. The magnitude of change is then determined by comparing the feature vectors using the Euclidean norm. According to the hypothesis, comparing two images when there has been no obvious change would lead to similar vectors in the feature vector's dimensional space.

A Contrastive Loss function is used to train Siamese networks. It attempts to reduce the distance between the two feature vectors if there is no change and vice versa. The illustration that follows makes an attempt to illustrate it using a scenario involving two classes. If neither of the two photographs showed the modifications we were practising for, the real label would be 1.

The strengths of this algorithm are Well Paired With the Best Classifier, Learning from Semantic Similarity, More Resistant to Class Inequality. Weakness of this method are requires longer training than typical networks, Avoid Producing Probabilities.

5 Result Analysis and Observations

From the Google Earth Engine, satellite pictures are downloaded. It provides us with multispectral images of Switzerland. Images with two different time stamps were acquired in order to identify the changes that have taken place over the past few years. The output of Remote Sensing Change Detection (RSCD) is a binary image that clearly identifies the locations of Land Use and Land Cover (LULC) changes.

The following images show the satellite image at time stamp T1 i.e., (a), the satellite image at time stamp T2 i.e.,(b), and the third image is the change map we obtained i.e.,(c)



Fig 5. The changes occurred in the agricultural land due to urbanization

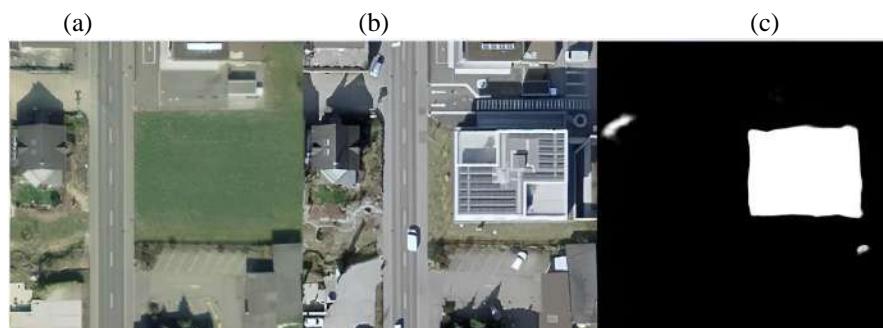


Fig 6. The changes occurred in agricultural land due to construction of building

The Siamese neural network generates accurate results. The Confusion Matrix is created via the suggested approach. A table that shows how many predictions a model got right and wrong is called a confusion matrix. It is used to assess the efficacy of a categorization model. When the expected value and the actual value are both positive, TP takes place. When the actual and projected values are both negative, TN happens. When an optimistic prediction is made but the actual is adverse referred to as the Type 1 error as well. FN is the outcome when both the fact and

the prediction are true referred to as the Type 2 error as well.

Table 1. Confusion Matrix

	Change	No Change
Change	90	1
No Change	12	106

The Siamese neural network produces the metrics shown below. The basic metric for model evaluation is widely used to measure accuracy, which counts the percentage of accurate forecasts over all other predictions. Precision is measured by the frequency of correct positive forecasts (true positives). A measurement that combines recall and precision is the F1-Score.

Table 2. Metrics

METRICS	FORMULA	VALUES
ACCURACY	(TP+TN)/(TOTAL)	0.938
PRECISION	(TP)/(TP+FP)	0.989
RECALL	(TP)/(TP+FN)	0.882
F1- SCORE	2(PRECISION*RECALL)/ (PRECISION + RECALL)	0.932

- F-Measure offers a single score that integrates precision and memory problems into a single value.
- It was noted that a binary image was generated, with the white portion corresponding to the change and the black portion corresponding to no change.
- It was found that the model had a 93.8% accuracy rate.
- Precision counts the number of positive class predictions that really fall into the positive class. The model's accuracy was found to be 98.9%.
- Recall quantifies the number of times the dataset's valid examples were used to make accurate class predictions. The model's accuracy was demonstrated to be 88.2%.

6 Conclusion

In this project, Switzerland was chosen as a specific place from Google Earth Engine, and we utilised two different timestamps to check for changes. These photos, which were taken using multispectral remote sensing technology, were used in our investigation. We used a Siamese neural network to achieve change detection of the input location. Through the Siamese neural network featured vectors of the two input images are calculated and the euclidean distance between the two featured vectors are computed. In the change map the white color denotes the change and

the black color denotes there is no change. Several acquisition functions and methods for calculating prediction uncertainty have performed admirably in our tests. All qualified strategies significantly outperform a naive random baseline and accomplish the performance of a model that has been trained using only a small number of samples from the entire collection of available data.

7 Future Scope

The project's objective is to identify changes in agricultural areas using remote sensing photographs. Consequently, there is a chance for change in agricultural areas. Our future research will include calculating the percentage change and percentage change in water bodies, mountains, before and after natural disasters, and other characteristics.

8 References

- [1] P. Coppin, I. Jonckheere, K. Nackaerts, B. Muys, and E. Lambin, “Digital change detection methods in ecosystem monitoring: A review,” Int J. Remote Sens., vol. 25,no. 9, pp. 1565-1596,2004.
- [2] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, “Deep Learning in remote sensing applications: A meta-analysis and review,” ISPRS J. Photogramm. Re-mote Sens., vol. 152, pp. 166-177, Jun. 2019.
- [3] L. Zhang, B. Du, “Deep learning for remote sesing data: A technicaltutorial on the state of the art,” IEEE Geosci. Remote Sens. Mag., vol.4, no. 2,pp. 22-40, Jun. 2016.
- [4] K. L De Jong and A. S. Bosman, “Unsupervised change detection in satellite images using convolutional neural networks,” Dec 2018.
- [5] N. Kadhim, M. Mourshed, and M. Bray, “Advances in remote sensing applications for urban sustainability,” Euro-Medit. J. Environ. Integr., vol. 1,no. 7, pp. 1-22, 2016.
- [6] L. Wen, M. Matsuoka, B. Adriano, “Damage detection due to the typhoon Haiyan from high-resolution SAR images”, in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 3, no. 1, pp.123-133, May 2019, doi: 8.1207/MGRS.2019.2931830.
- [7] R. J. A Tough, D. Blacknell and S. Quegan, “A statistical description of polarimetric and interferometric synthetic aperture radar”, in IEEE Journal of

Selected Topics in Applied Earth Observations and Remote Sensing, vol.7, no.1, pp.67-93, April 2019, doi: 6.1107/MGRS.2019.1431430.

[8] Kevin Louis de Jong, Anna Sergeevna Bosman, “Unsupervised Change Detection in Unlabeled Optical Remote Sensing using CNN”, in IEEE Transactions on Geoscience and Remote Sensing, vol. 2, no. 1, pp.102-114, March 2019, doi: 6.1219/MGRS.2019.1214110.

[9] F. Bovolo and L. Bruzzone, “A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain”, IEEE Trans. Geosci. Remote Sens., vol.45, no. 1, pp. 218-236, Jan.2007.

[10] L. Wen, M. Matsuoka, B. Adriano, “Damage detection due to the typhoon Haiyan from high-resolution SAR images”, in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 3, no.1, pp.123-133, May 2019, DOI:8.1207/MGRS.2019.2931830.

Real Time Sign Language Detection Using OpenCV

Pavan Kumar Meka¹, Yesu Raju Parusu¹, and Radhesyam Vaddi²

¹Student, Department of IT, VR Siddhartha Engineering College,
JNTU-Kakinada, Vijayawada, India.

¹Pavanmeka74@gmail.com, ¹vesurajuparusu@gmail.com

²Assistant Professor, Department of IT, VR Siddhartha Engineering
College, JNTU-Kakinada, Vijayawada, India.

²syam.radhe@gmail.com

Abstract. Sign language is communication commonly used by people who are deaf and dumb. They use hand gestures to exchange information between their community and with the remaining people in the world. Real time sign language detection is the solution that provides the communication between normal people and those who are deaf and dumb. It deals with image acquisition and continues until text generation. Previously provided solutions are all sensor based and they don't give good accuracy. The old models customized with limited gestures. In order to overcome this problem, a model has been developed to detect the 40 different classes of signs. Which includes alphabets, numbers and some special symbols. This model can effectively recognize gestures dynamically. The dataset is created in well planned manner to include 10 numbers, 24 alphabets and 6 special symbols. The sign language recognition steps are in this survey are Data collection, image preprocessing and segmentation, training model and last classification and recognition of result. The captured images are preprocessed to remove the extra background in image. A convolutional neural network algorithm is used to train the model and perform multi-class classification on image data. The results are displayed in the monitor with text recognition. Some future improvements for research in this area also suggested.

Keywords: Sign language, classification, convolutional neural networks, hand gestures, communication, real-time.

1 Introduction

Communication is very important for sharing information and to express oneself. Deaf and dumb people use sign language to express their ideas and thoughts with their community and with remaining world. Sign language is actually a hand gesture that convey the meaning message. There are around 450 different classes of sign

languages are used globally. And in India there are around 250 certified sign language interpreters for deaf people. This is very difficult to teach the sign language for all the deaf and dumb people as there are limited number of interpreters available today.

To overcome this problem, a model is to be developed to detect the signs automatically. Previously developed models are all sensor based and they need gloves and high-resolution cameras to capture and identify the hand gesture. They are all limited to certain hand gestures only. Some models are able to detect only statics images, when it comes to dynamic images they do not work properly. These are all drawbacks of the previously developed models.

For images convolutional neural networks is the most popular neural networks than the other one. The used convolutional neural network algorithm (CNN) is a 3-layer network. The network layers are convolutional layer, pooling layer and fully connected layer. Datasets that we have created includes 40 different classes which are 24 alphabets, 9 numbers and 6 special symbols. With the help of deep learning model, we can easily identify the gestures. So, this system will help deaf and dumb people to communicate easily with remaining people in the world.

2 Literature Study

Here is the study of various previously available models, methodologies and techniques to sign language which were referenced to create our proposed model.

Hand Gesture Recognition Using PCA (2020) [1]:

Authors mainly concentrated on the static hand sign gesture detection system using image processing. For hand signs a vector called SIFT algorithm is used. Proposed a model to increase the communication of deaf and dumb people with other communities and based on old traditional model. Model not works well for dynamic signs and gives output with less accuracy.

An Automated System for Indian Sign Language Recognition (2018) [2]:

Developed a method for automatic recognition of signs on the basis of shape-based features is presented. Features of segmented hand gestures are calculated using Hu's invariant method that are given as input to Artificial Neural Network for classification.

Design of ANFIS system for recognition of single hand and two hand signs (2019) [3]:

Author developed a model that helps the deaf and dumb people to communicate with the rest of the world using hand gesture detection. The important thing in this system is conversion of real time hand sign into text format. They have used the neural networks and histogram of gradients (HOG) for building their model and it has the capacity to recognize up to 20 signs that contains the mixture of both numbers and some alphabets.

Finding relevant image content for mobile sign language recognition: Signal Processing, Pattern Recognition and Application [4]:

This work deals with deep learning algorithms like CNN, RNN which work better on image data. The algorithms like CNN has better ability and better understanding of the image. And performs the better classification among different classes of signs. But the accuracy for this model is not up to the mark. It works well for the static images.

Learning to Estimate 3D Hand Gesture from RGB images [5]:

In this paper the author mainly concentrated on the estimating 3D Hand gesture from single RGB (Red Green Blue) or Colored images. And finally, he developed a deep model that understands or recognizes the network 3D articulation. The main drawback is its ambiguities because of blank depth details.

Vision based hand gesture recognition. (2021) [6]:

He proposed a real time vision-based system for hand gesture recognition for human computer interaction in many applications. The system can recognize 35 different hand gestures of Indian and as well as American Sign Language with less accuracy.

American Sign Language Translator [7]:

The author has researched 5 years to build an automated sign language translator. And he finally developed a model based on convolutional Neural Network (CNN) and Support Vector Machine (SVM). These two algorithms work well for comparing images and performs better classification. The standard video camera is used for obtaining data. Obtained data is served as input for the model.

Fingerspelling detection in the wild with iteration visual attention [8]:

This project focuses on fingerspelling patterns in American Sign Language (ASL) videos gathered from various resources like social media, YouTube and many other wild life channels. They proposed a model based on an iterative mechanism.

Sign Language detection via Skeleton-Aware Multi-Model [9]:

The authors developed a model that extract main features in the picture via deep neural networks. It performs well in detecting body postures, hand signs and facial expressions also. Drawback is this model suffers a lot when there is an inadequate and noisy data

Neural Sign Language Translation [10]:

Here authors developed a model that generate the spoken language from videos that contains sign languages. It continuously detects sign but only drawback is it ignores the linguistic and grammatical structures of sign language.

Sign Language generation, recognition and translation [11]:

This project requires information about various fields like computer vision, natural language processing, deaf culture and computer graphics. And results produced by this model was displayed at workshops. They have used a CNN model in this project.

A comprehensive study on Deep learning model for sign language detection [12]:

Here authors conducted a comparative analysis of computer vision-based methods especially for sign recognitions. And this entire study is focused on giving insights on sign recognition and mapping non-segmented streams of videos. And this also provides the speech to text recognition.

Hand Gesture Recognition using Image processing and feature extraction techniques [13]:

This paper is mostly focuses on use of traditional preprocessing techniques to detect the correct hand gesture effectively. k-NN (K-nearest neighbor) is the algorithm used in this model. Basically, this algorithm stores all the data and classifies them based on their highest similarity. And this model is works well for static images. Whereas in the case of dynamic images model do not work properly.

3 Proposed Work

The proposed model uses the OpenCV to read and capture the hand gestures using webcam. After that we will pre-process image to remove the extra background from the image so that model will focus on main part of the gesture. A model is trained using Convolutional Neural Network (CNN) to classify the different classes of hand gestures. Convolutional Neural Networks works well for the images rather than any other neural networks. It classifies the hand gestures with more accurate and effectively whenever gestures are shown to webcam.

3.1 Data set

The proposed model was trained on 8000 colour images of different classes of hand gestures. Which includes 26 alphabets, 9 numbers and 6 specials symbols or English Phrases. For this model we have created our own data set by capturing different classes of hand gestures in different angles and light adjustments.

Table. 1. Dataset Structure

Gesture type	No. of data samples
Alphabets (A-Z)	Each gesture contains 200 images $26*200 = 5200$
Numbers (1-9)	Each gesture contains 200 images $9*200 = 1800$
Okay	200
Call me	200
Rock	200
Smile	200
Thumbs down	200
Thumbs up	200

The table shows gestures type and no.of copies made for each type. Actually this project runs on mainly two different programs one for data creation and another one for testing model. We will create datasets required for our project using data creation program. For each sign we have created 200 images at different angles. And with same dataset we are going to train our model. The type of the gestures above mentioned are alphabets, numbers and 6 specials like call me, Rock, Thumbs up etc... so at last we have 8000 images of different classes of hand symbols to train the model

3.2 Pre-processing

The captured images were pre-processed to remove the additional background in the hand gestures by using image filtering and Segmentation. So, this will modify or enhance image properties and extract the valuable information from the pictures such as corners, edges and etc... A filter is nothing but kernel or some particular pattern, which is applied to each pixel and its neighbours within an image.

A total of 200 images were pro-processed to enhance the image properties.

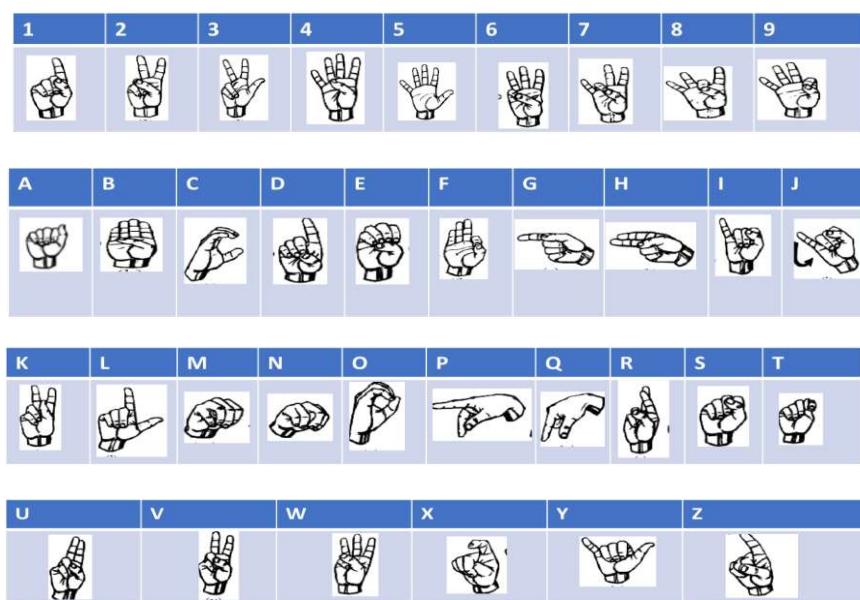


Fig. 1. Created dataset

These are all different hand signs that we have created and we also mentioned what is their respective meaning in the sing language. For example you can see for the digit 1 symbol or sign is showing index finger. So we have created dataset for all alphabets, numbers and some special symbols. We have taken all this hand signs from indian sign language (ISL).

3.3 Design Methodology

This model mainly uses the Convolutional Neural Networks (CNN) and TensorFlow (as Keras uses the TensorFlow as their backend) for training purpose. The design methodology of the proposed model has been described below:

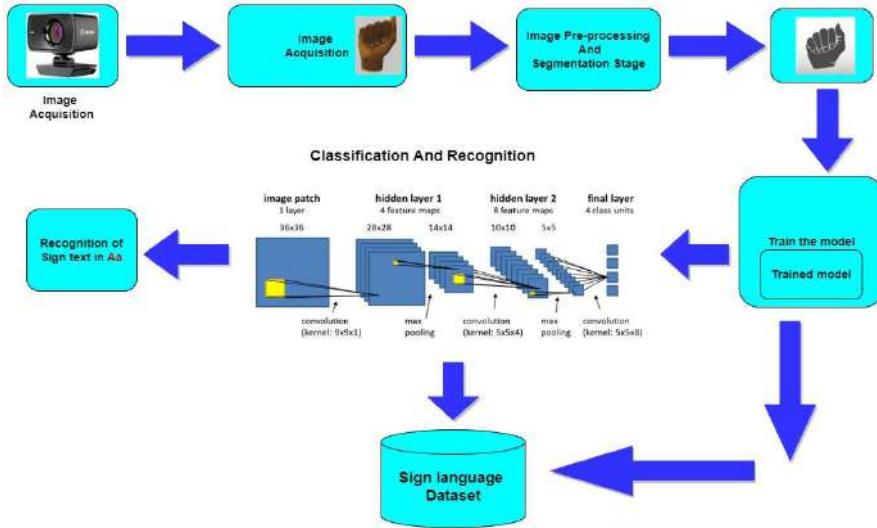


Fig. 2. Architecture Diagram

The first step is we have captured hand gestures of different classes using different angles. After that each image is pre-processed to enhance the image properties. And all the images are stored in database with labelling. Then, these images are used to train the model. A CNN model was created by including three main layers named convolutional layer, pooling layer and fully connected layers.

This deep learning model was completely trained and tested using our own dataset. Once the we got the required accuracy, the model has been used to make predictions of the hand gestures.

3.4 Extract Hand Gestures from pictures

As our project is sign language detection, we focus more on hand gestures. The inputs for the project is taken with the help of webcam. But when it comes to the model it needs only valuable information (hand gestures) from the given picture to detect a particular sign. When we are running the model two outputs will be displayed. One is our original pictures and other one is white-image also called as pre-processed image. In the pre-processed image extra background will be removed and some rays will be added to the hand gestures which are the main base for the classification.

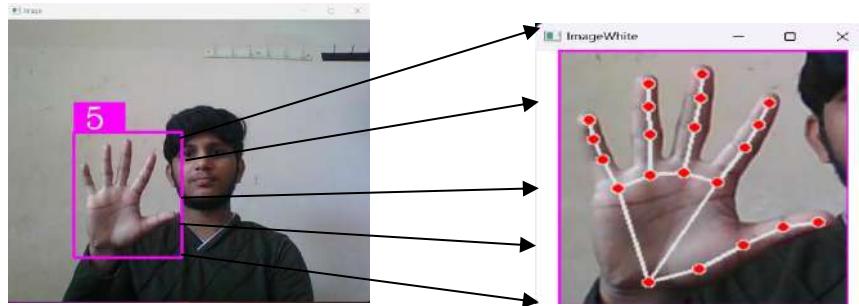


Fig. 3. Extraction hand gestures from pictures

The above picture shows that whatever the picture is taken by the webcam it will be furtherly modified and only hand gesture part will be given to the model as an input. We will this process in image pre-processing where background of the picture will be removed and enhancement of the picture takes place. And finally all the pre-processed images will be used for model training purpose.

Algorithm:

Step 1: Create a separate file for dataset in project folder.

Step 2: Create required dataset by capturing images of hand gestures using webcam.

Step 3: Add labels to each image and store them in a database (separate file)

Step 4: Perform image pre-processing to each image that is captured.

Step 5: Build a CNN (convolutional Neural Networks) model using our own dataset.

Step 6: Train and test the model using customised dataset.

Output image using convolutional layer = $N-F+1$

Output image using pooling layer = $N+2P-F+1, S=2, P=2$

Where n= size of input, F= size of kernel/filter, P=padding, S=stride

Step 7: Use this model to **detect the sign** language

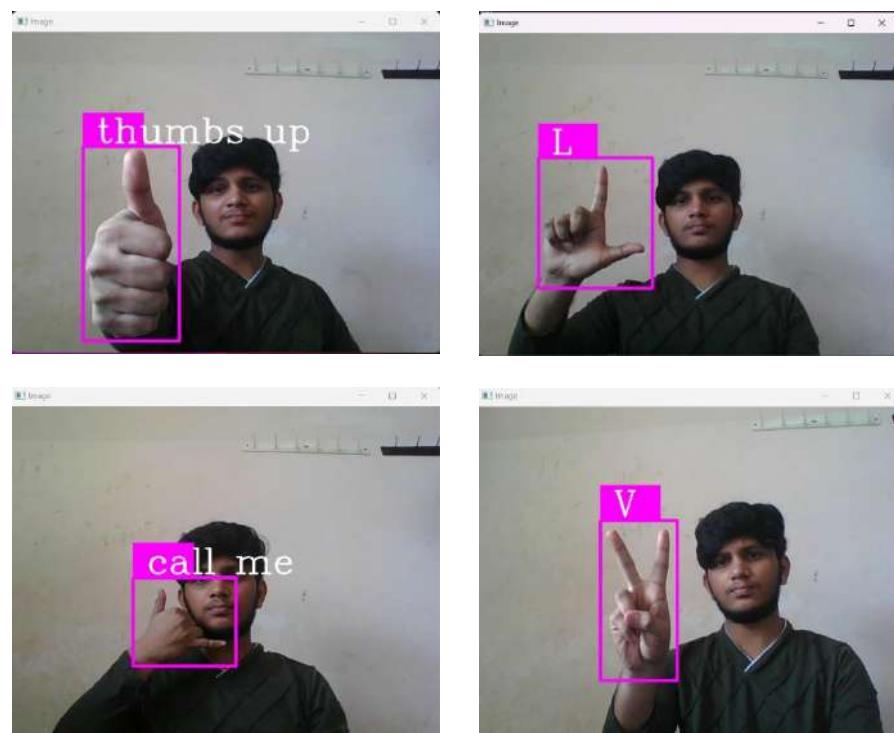
Step 8: Use the webcam to test the trained sign language recognition model

The algorithm used for this model is **Convolutional Neural Networks (CNN)**. There are many neural networks and algorithms for detecting various signs and gestures. But, CNN has better ability to **compare images** and performs the best in **classifying the various types of hand gestures** than any other neural networks algorithm. That's why we have choosen CNN algorithm for our project. And this CNN mainly contains three layers. They are Convolutional layer, Pooling layer and fully connected layer. These three layers are core of the CNN and during this

process the input image is passing through these layers. In each layer the input image is converted into specific format. That will be server as a input for the next layer in the CNN. The formalas and mathematical expressions of these layers are mentioned in the above algorithm format. We have used teachable machine for building our model with our own dataset. Based on our datasets the teachable machine will build a machine learning model and we have used this model for testing purpose inorder to detect the sign. The inputs for the model is given with the help of webcam so that it will **detect sign** and displays its correct meaning as a output.

The trained model gives the accuracy of overall 90.56%.

4 Results and Observations



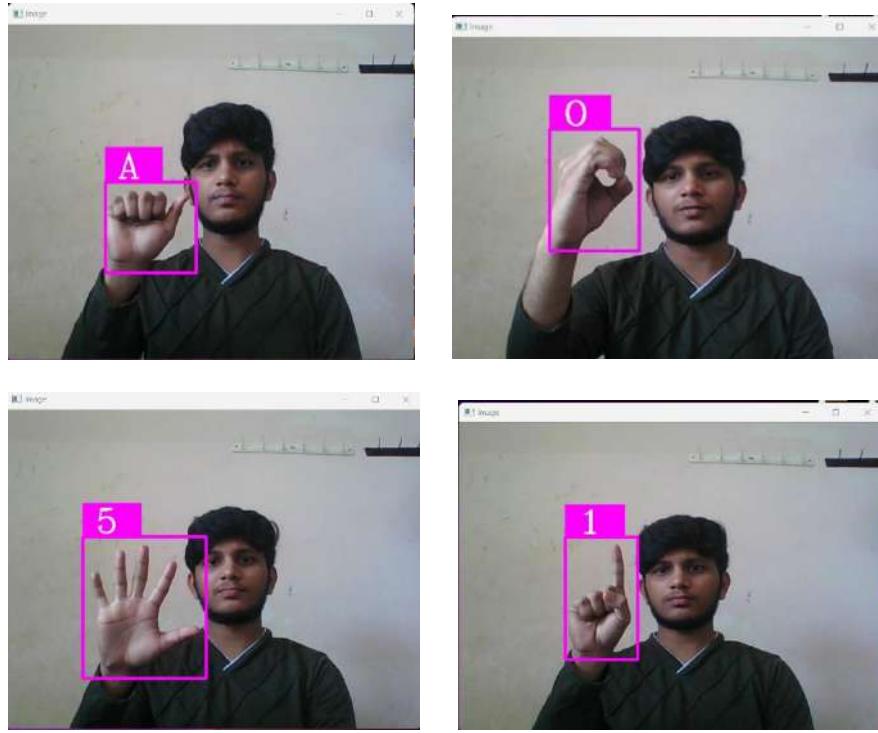


Fig. 4. Outputs produced by model

The above figures are outputs that produced by our model. After training the model, we have tested it with different signs at different angles.

We did this model for 40 different signs and some signs have almost same hand structure for example symbol “2” and symbol “v” have almost same hand structure. Due to this reason the model sometimes produces the incorrect outputs.

Table. 2. Interpretation standards of the model

Interpretation Standards	Proposed model
Precision	0.89
F1-score	0.85
Recall	0.91
Support	91.62
Testing Accuracy	90.56
Weighted average	89.43

The model was executed for 20 spans until required accuracy was obtained. The Interpretation Standards like precision, recall and F1-score of the model are displayed in the above table. These are all various output parameters of our model. The main accuracy we got for this model is 90.56%.

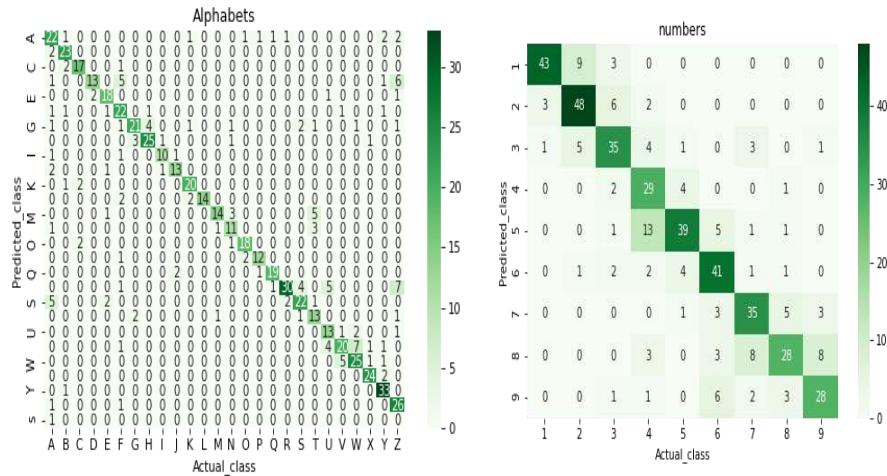


Fig. 5. Confusion Matrix

These are confusion matrices produced for both alphabets and numbers type of hand signs. As you can see in the most of the times model detects the correct sign and that is indicated with dark green boxes in the confusion matrix. There is a diagonal line the confusion matrix. In this line predicted sign is same as actual data or sign. This indicates that if there is a high dark boxes in the daigonal means the model detection accuracy is high. Otherwise it will results in low accuracy. There is a small deviations from dark green boxes which shows that incorrect sign detected by the model due to ambiguity of same hand structures of signs. Sometimes the model's detection maybe false in that cases the confusion matix shows the some deviations from dark green daigonal boxes. But the model executes exceptionally well as the training progresses. The validation set loss is seen to fluctuate heavily during the entire training period.

The testing accuracy of 90.56%, precision of 0.89, recall of 0.91 and an f1 score of 0.85.

5 Advantages

- i. It helps the children who have autism spectrum disorder (ASD).
- ii. It gives the literacy for Dumb and deaf people.
- iii. Elderly people benefited greatly from the use of sign language detection model.
- iv. Sign language model gives efficient and accurate way to convert sign language into text in real time.

6 Conclusion

In this report, A Real Time sign Language detection model have been developed to recognise various signs and finally helps deaf and dumb communities to interact with remaining people in the society. And this model plays an important role in the society to help the people who are suffering with autism spectrum disorder (ASD). The aim of the model to recognize various sign language signs and transform them into text. It gives high accuracy of 90.56%, precision of 0.89 good response time when applied on practical scenarios. There are many traditional models available for this project in the past. But they are all sensor based and they do not work well in all the cases. And sometimes they restricted to only static hand gestures. But our model can detect upto 40 different signs that includes Alphabets, Number and even some special signs. In future we can develop apps on these models so it will help deaf&dumb children in their studies. It can also be deployed as an API using cloud such that it can be integrated with other applications easily and finally maintains a good communication among deaf&dumb communities.

7 Future Scope

1. we can make this project as mobile application so that people can easily understand sign language using their mobiles. Mobile camera will be used as input for the model and model will recognize hand sign and put the recognized result in the mobile screen.
2. We can develop a model that directly converts sign language into voice. We need to attach speaker to the model. And the outputs will come from the speaker (voice form) when model recognizes that particular sign.

References

- [1] Agrawal, S. C., Jalal, A. S., & Tripathi, R. K. (2016). A survey on manual and non-manual sign language recognition for isolated and continuous sign. International Journal of Applied Pattern Recognition, 3(2), 99-134.
- [2] Cheok, M. J., Omar, Z., & Jaward, M. H. (2019). A review on hand signs and sign language detection techniques. International Journal of Machine Learning and Cybernetics, 10(1), 131-153.
- [3] Kausar, Sumaira, and M. Younus Javed. "A survey on sign language recognition." In 2011 Frontiers of Information Technology, pp. 95-98. IEEE, 2011.
- [4] Rastgoo, Razieh, et al. "Sign language production: a review." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.
- [5] Ahmed, Mohamed Aktham, et al. "A review on systems-based sensory gloves for sign language recognition state of the art between 2007 and 2017." Sensors 18.7 (2018): 2208.
- [6] https://www.youtube.com/watch?v=wa2ARoUUdU8&t=2547s&ab_channel=Murtaza%27sWorkshop-RoboticsandAI
- [7] Aloysius, Neena, and M. Geetha. "Understanding vision-based continuous sign language recognition." Multimedia Tools and Applications 79.31 (2020): 22177-22209.
- [8] World-wide deep sign language detection from Video: A large datasets and Methods comparision by Dongxu Li, Cristian Rodrigues (2019).
- [9] Ghanem, Sakher, Christopher Conly, and Vassilis Athitsos. "A survey on sign language recognition using smartphones." Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments. 2017.
- [10] Viswanathan, D. M., & Idicula, S. M. (2015). Recent developments in Indian sign language recognition: an analysis. International Journal of Computer Science and Information Technologies, 6(1), 289-293.
- [11] Jiang, Xianwei, et al. "A survey on artificial intelligence in Chinese sign language recognition." Arabian Journal for Science and Engineering 45.12 (2020): 9859-9894.
- [12] <https://towardsdatascience.com/sign-language-recognition-with-advanced-computer-vision-7b74f20f3442>
- [13] Mohandes, M., Deriche, M., & Liu, J. (2014). Image-based and sensor-based approaches to Arabic sign language recognition. IEEE transactions on human-machine systems, 44(4), 551-557.
- [14] Cooper, H., Holt, B., & Bowden, R. (2011). Sign language detection. In Visual analysis of humans (pp. 539-562). Springer, London.
- [15] A Review on Sign Language detection for the deaf and dumb Community by R Rumana, Reddygari Sandhya Rani, Mrs. R. Prema (2009).

Social Media as the Most Effective Means of Business Promotion Today With Using Social Media Advertising Technology

Aldo Saputra¹, Ingwen Tannaris², Kelby Hubert³, Ford Lumban Gaol⁴, tokuro Matsuo⁵

¹Bina Nusantara University, Information Systems Department, Jakarta, Indonesia

²Bina Nusantara University, Information Systems Department, Jakarta, Indonesia

³Bina Nusantara University, Information Systems Department, Jakarta, Indonesia

E-mail: ¹aldo.saputra1@binus.ac.id, ²ingwen.tannaris1@binus.ac.id .com,³

Kelby.Hubert1@binus.ac.id, ⁵fgao@binus.edu, ⁵matsuo@aiit.ac.jp

ABSTRACT

Indonesia has the largest number of people that are active in social media and Indonesia is a country with the most social media users in the world. Social media in general is used to socialize (relate, both, personally, in groups and so on) between users. The users of social media in Indonesia already has reached 160 million in January 2020. and this number of social media users in Indonesia increased up to 12 million (8.1 percent) from April 2019 to January 2020. Firstly social media was used by users to connect with each other through community. However now social media is not only used as a means of virtual community for interaction but is also used as a channel for doing business. There are still many people who do not realize that social media has the advantage of being a very efficient and effective business, and many company that still confused about to choose which social media platform that suitable with their business promotion. Therefore we want to show and tell them one of the advantages of social media in the field of promotion. Based on the surveys there are four most popular social media in Indonesia such as : Youtube, Whatsapp, Facebook, and Instagram. However there still very few studies can be found to understand that social media is effective or not for business promotion. For this reason, this article will examine the advantages that social media have that can support business promotion. This article will utilize the questioners method to assess the important factors of social media users such as : user satisfaction factors in using social media, what social media are often used by user to promote their business, etc. The research result are expected to provide valuable guidance for user to understand about function social media for business promotion.

Keywords: *Social Media, users, Trust, Satisfaction*

1. INTRODUCTION

Today social media has been used by many people in Indonesia because it's fun to see other people's activities, watch some funny or cool stuff, learn something from social media, and many more. The more fun stuff we can do in social media, the more social media becomes popular and the more effective for social media to promote products or businesses [1].

Many big companies have realized that social media is the best way to promote a product or businesses. Many big companies used social media to promote their products or businesses because they know that they can reach people from very far away from them very easily [2]. However there are still many companies who didn't use the advantage of social media to promote their business because they didn't know. Also many companies nowadays confuse choosing which social media platform they think is suitable to use for their business promotions tools. Therefore we are now doing research about what social media are currently in great demand and use by the public. From the research we will show what are the benefits that can be obtained from using social media as a means of doing business and which social media that suitable for use as a means of promotion

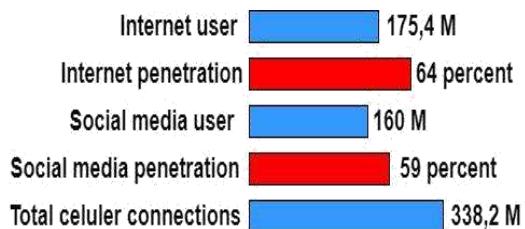


Figure 1. Number of internet users and social media users[1]

We got a result after doing research on the internet and we found the fact as we can see from the graphic above it explains that 90% of all internet users use social media. In Indonesia there are 267.7 million people based on population data, it means that social media users reach 60% of total population in Indonesia, and it's still increasing rapidly [3].

The Internet provides new ways to interact between humans. It destroyed walls that limit human direct interaction with other humans and changed human interaction patterns. Technology makes us don't need space and time to interact with other people [4]. The large number of internet uses and the high need for interaction make the use of social media very much needed, especially among young people. Companies must take advantage of this condition. Through social media companies can expand their business widely. Therefore, we are interested in researching how effective social media is for promoting business [5].

From all the explanations above, we conclude there are some questions that are most asked about social media from people, the question is like how effective social media impacts business growth? Which social media that support business promotion or have features for supporting business? Which social media type that is most suitable for business promotion?

2. LITERATURE REVIEW

2.1 Social Media

Social media is a technology that uses the internet to facilitate all people to share ideas, thoughts, and information to each other so all people can connect with each other. Social media used to interact with friends and family but by time social media used for businesses that want to reach many people effectively through this popular communication method that social media serves. Social media's role in helping businesses is significant. It made business easy to communicate with customers[6].

With the gathering information advantages we can improve our marketing and market research to another level. It's easier to promote our products. because we can determine our market target, time ,and duration perfectly. And then, with social media we can build customer relationships through customer's trusts in our social media account [7].

1. Social Media According to Antony Mayfield

According to Antony Mayfield who provides his ideas in the form of a definition of media social where according to him social media is a media with user convenience participate, share and create roles, on specific blogs, social networks, online wiki / encyclopedia, virtual forums [8]

2. Social Media According to Andreas Kaplan and Michael Haenlien

According to Andreas Kaplan and Michael Haenlien defining the meaning of social is a group of internet that have a foundation based with the internet. ideology and technology of Web 2.0 enable the user to create and exchange some content [9].

3. Social Media According to Lisa Buyer

According to Lisa Buyer, the definition of social media is a form of public relations that most transparent, attractive and interactive today [10]

4. Social Media According to Sam Decker

According to Sam Decker, social media is content digital and the interactions created by and between each other [11]

5.Social Media According to Marjorie Clayman

According to Marjorie Clayman the definition of social media is a new marketing tool that make it possible to know customers and potential customers in a way that was not possible before [12].

Types of Social Media

- Online Communities groups have come in all types and sizes, many of them are created by people or a group of people without any company affiliations. Many log or groups in social media were created for a specific purpose for example in line application there were many groups that were created by groups of people that used to help other people to search for a part time job. there also another group that have another purpose like a group for promotion their business, etc.
- Blogs, the users of the blog are very diverse, some blogs are personal to close friends and family, and there are also other blogs that are designed to reach and also influence a wide audience with their blog. there is also a blog that created to inform a information.
- Social networks have an important role in both consumer business and business-to-business marketing. One of them is from Facebook, Messenger, Twitter. These different networks have offered different benefits according to their platform.

2.2 Business

The first thing to make a business is assign the business concept. Depending on the business, Extensive market research may be needed to determine whether turning an idea into a business is viable, determining how to promote business products and whether the product has value for customers. a business must know how to give customers the best experience when seeing a business product and can make the customer have his best Satisfaction to buy the product [13].

2.3 Advertising

1. Advertising

Advertising is a way of communication with many people through various kinds of media that are held by companies, non-profit organizations, or individuals. The main purpose of advertising is to share our product or services to many people so it can increase our demand of product that being advertised [14]

2. Public Relation

Publicity is the total of information that is shared to many people through certain media about a person, item, or organization. Public relations, building good relations with the public is a branding effort very effective. Participating in social activities using company attributes also products have a positive value in the eyes of society. They (society) give positive assessment of your company, the most important of which is all "brand image" the company is getting stronger and stronger [14].

3. Sales Promotion

Sales promotion is marketing activity to encourage customers about the effectiveness or quality of an item through some demonstration, exhibition or discount.

2.4 The Most Popular Social Media that suitable for business

Advertising is a marketing tactic or method that involves paying for some space to promote a product, service, or cause. The actual promotional messages are called advertisements, or ads for short. The goal of advertising is to reach people most likely to be willing to pay for a company's products or services and entice them to buy. now advertising product can be everywhere like from television , newspaper or billboard but now mostly product are advertise at social media because most people active at social media.

1. Youtube

Youtube is a website or application that provides a free video sharing that can make it easy to watch online videos. Everyone can create and upload their own videos to share with others as creators. YouTube has become very popular because everyone can watch a video according to their likes, there are also many people that want to be creators on youtube because now we can get money from youtube by accepting the adsense according to your view in your video. We can also promote some business on youtube by adding ads in some videos.

The advantages of youtube for business :

- Marketing in youtube is very easy you only have to paid to youtube to add a adds in their video or you also can paid the content creator to promote your business
- Your content will never be deleted as long as it doesn't violate youtube policies.
- You can grow your audience worldwide by upload your video in youtube
- Your can grab the audience attention to your business using short video or adds

2. Instagram

Instagram is a free application that allows user to share their expressions through photo or video to other people.

Instagram allows people to edit and share their photo or short video through the application with some hashtag or geotags that make the post searchable by other people. Each post that is made by a user will appear on their follower's feed and public when the post is tagged with hashtag or geotags.

The advantage of instagram for business :

- connect with customers across multiple channels
- Reuse marketing materials
- Attract engaged traffic with a community
- boost your marketing with ucg like photos from customers

3. METHODOLOGY

3.1 Research Model

Table 1: Research variables and indicators

No	Variable(s)	Indicator(s)	Reference(s)
1	Active user	total of use using social media	[3]
2	Popular social media	Social media mostly use by people and have most active user	[4]
3	Advertising type	Type of advertising in	[5]

		the social media, example video ads, picture ads and more	
4	Advertising target	Target audience with certain traits	[7]
5	Active ads	Total of the ads show to user in one day	[8]

3.2 Data Gathering Method

The method used in this study is questioner, where this questionnaire contains some questions that are related to social media, the respondent will answer according to their opinion and feelings about social media. the questionnaire also will ask about how social media can be used as a means of business promotion.

Here are some of the questions we ask to respondents :

- How often do you use social media
- Social Media is used most often
- Where you see most ads
- Have you ever been interested in advertisements on social media?
 - If you have a business, what media will you choose to promote?
 - In your opinion, is social media effective for promotion? Why?

This study uses phenomenology method because this research requires observing the reality that occurs daily in human life. This research will observe the reality through questionnaires that will be shared to many people. We can dig up data from people's answers through the questionnaires and draw conclusions from it.

3.3 Data Analysis Method

In this research we used google form technology to collect respondents that we distributed via the line application. We get respondents as many as about twenty people that we will use to draw a conclusion. This research shows that social media is very widely used by people every day so that social media is not only a place to express yourself but also as a place to promote a product that is very effective today.

This study uses a qualitative phenomenological method. Phenomenology word Derived from the Greek word, phainomeno which means appearance of oneself and logos which means reason, phenomenological study is a study that specializes in visible phenomena and reality to examine the explanations in them. Phenomenology itself has two meanings, namely as a philosophy of science and also a research method, which aims to find the meaning or meaning of the experiences that exist in life.

4. RESULT AND DISCUSSION

4.1 Chart of Age of Respondent

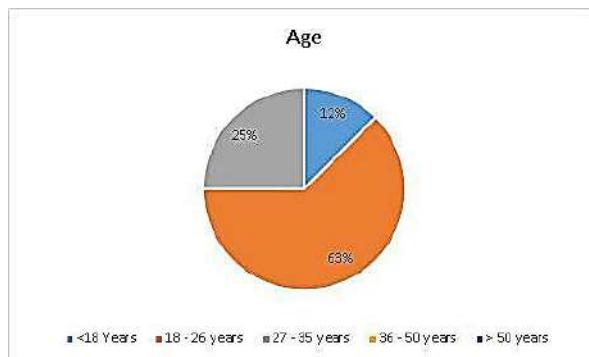


Figure 2. Range of Age

From the Figure 2, it is found that most respondents are 18-26 years old who are adolescents to early adulthood. We can conclude that the range of age from millennium ages.

4.2 Chart of Social Media Users

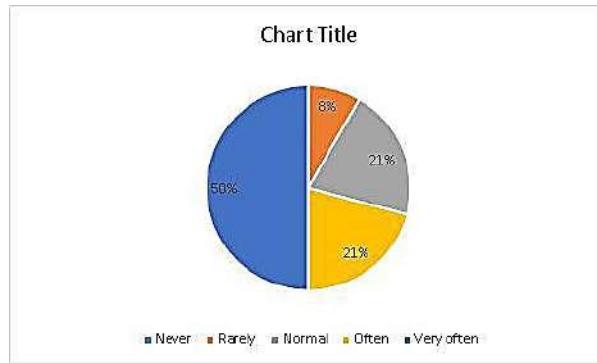


Figure 3. Social media users

From the Figure 3, we can see if 50% of users are very often using social media, 25% often, 15% ordinary and 10% rarely. From this chart we can know how often users use social media.

4.3 Chart of Social Media That Used by User

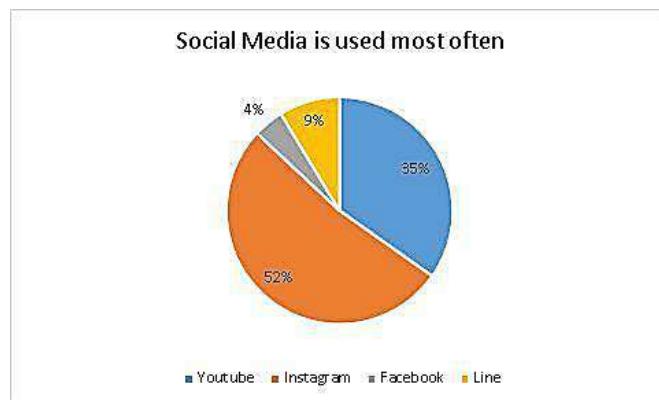


Figure 3. Social media users

From the Figure 3, we will get the information of social media that is often used by users, in this question we give 5 options of social media such as youtube, instagram, facebook, line, and telegram. We can see if instagram is the most widely used by users by 50%, second is youtube by 30%, third is line by 10%, and last facebook and telegram by 5%.

From this information we can know what social media that often use and being seen by users so we can analysis what social media that suitable for business

4.4 Chart of Advertising

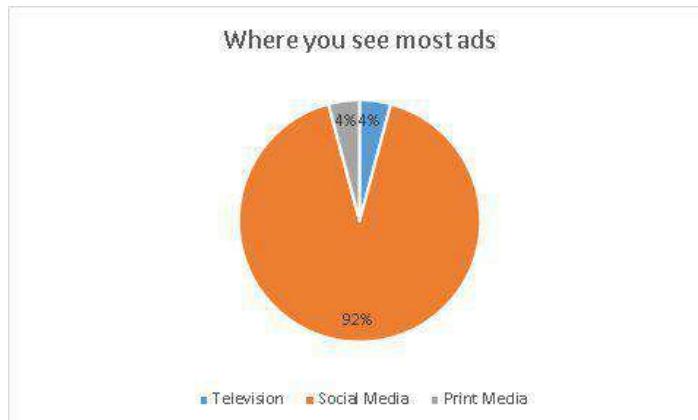


Figure 4. Adds on the Social media users

Figure 4 shows that users frequently view adverts in social media by 90% since social media is now widely and easily used by users. In social media, we may see a lot of information and advertise. We can now see social media on a variety of platforms.

4.5 The Respondent Interest from Ads

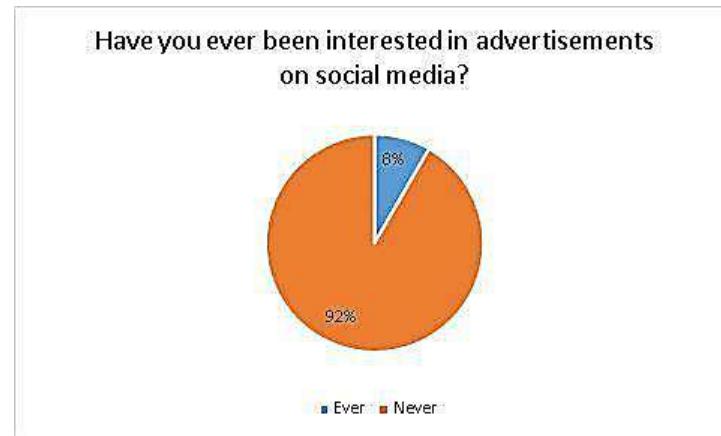


Figure 5. The Respondent Interest from Ads

According to Figure 5, the majority of respondents were interested in the advertisements; 90% were interested in the product and 10% were not. This leads to the conclusion that social media advertisements can attract more customers.

4.6 Chart of Best Social Media for Promotion Business

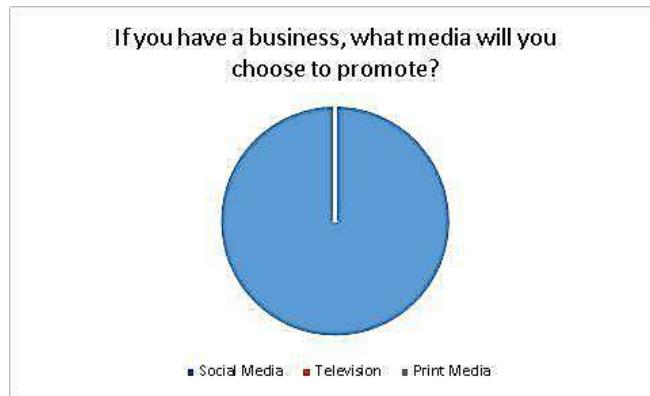


Figure 6. The Respondent of Social Media for Promotion Business

According to Figure 6, all respondents chose social media as a venue to promote business products, implying that social media is extremely effective for promotion. We might conclude from this experiment that social media has become an alternative for advertising. Social media has become the most apparent way to advertise in the digital age.

4.7 Chart Rate about How Often User Using Social Media

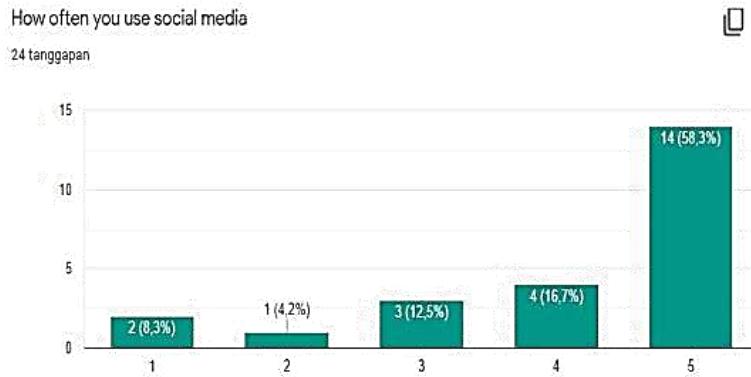


Figure 6. The Respondent of Social Media for Promotion Business

Figure 6 depicts the rate of social media users, with 21 respondents, the majority of whom use social media frequently, and only two respondents who use it infrequently.

4.8 Chart of User Who have Ever using Social Media to Promote

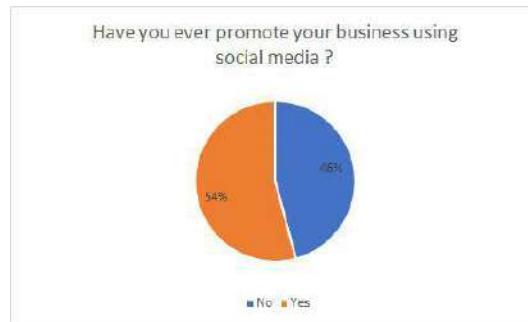


Figure 7. The Respondent of using Social Media to Promote.

From Figure 7. 57,1 % users ever use social media to promote their business, and 42,9 never. There is possibility why some people in this chart don't use social media for their business, is because all respondent who answer this questionnaire are in 18 -26 ages, so it's likely that not everyone in this age owns their business.

5. Discussion.

Here are the result answer of respondent for our research with using google form:

- Very effective, because many people are now more engaged on social media, especially when they have downtime. An appealing social media presence piques the user's curiosity.
- Very effective, because now all social media platforms that offer free services, such as YouTube, will undoubtedly catch adverts every time a video opens, which is very useful in attracting new users.
- Very effective, because now many people have social media and cannot be separated from it, especially since everyone uses social media 3-4 hours a day

It is particularly effective since everyone uses social media, and with social media promotion, people learn about the promotion more rapidly. Because individuals are more active and use the app more frequently in today's period, promotions can be delivered more effectively on Instagram and YouTube. Effective because youngsters are now more focused on social media than those who look around to be interested in the promotions supplied since social media is available where everyone who uses the internet can access and see it.

Very effective since practically everyone nowadays utilizes social media, and its dissemination is rapid and extensive. Social media is a powerful tool for promoting a product, especially if it is new. Because social media is so widespread and easy to distribute information on.

For all respondents, the majority responded that social media is extremely useful for product promotion since it has a large number of active users, is easy to share, and attracts new customers.

5. CONCLUSION

Most individuals spend their time interacting with social media, and some cannot get away from it because they believe it is a part of their lives. Because most people are engaged in social media, most people will view all content in social media that includes commercial product advertising. People use a variety of social media platforms, including Facebook, Instagram, Whatsapp, YouTube, and others.

Instagram, Youtube, and Line are the most popular social networking platforms. It is possible to infer that Instagram, YouTube, and Line are the most successful social media platforms for business promotion. Furthermore, individuals today are far more interested in social media than they are in television and print media. Young people also prefer to advertise their businesses on social media rather than other forms of media since social media is thought to be more successful and efficient in its execution.

As a result, social media might be considered the most effective promotional medium. According to our research, the type of social media that is best suited for business promotion is using a video or a picture, such as platforms YouTube and Instagram. YouTube uses a short video in ads to promote the business, and Instagram uses a short video and a picture of the product, as well as a link for users to get through the product.

REFERENCES:

- [1] Bimo, P. (2017, October 21). Perkembangan Media Sosial di Indonesia. Retrieved December 04, 2020, from <https://pakarkomunikasi.com/perkembangan-media-sosial-di-indonesia>
- [2] *Media Sosial, Tak Sekadar Jaringan Pertemanan – Bebas Akses. Bebas*

- Akses. (2020). Retrieved 3 December 2020, from <https://bebas.kompas.id/baca/riset/2020/06/17/media-sosial-tak-sekadar-jaringan-pertemanan/>.
- [3] *What is an Active User? How does DAU and MAU fit in? | Adjust.* Adjust.com. (2020). Retrieved 4 December 2020, from <https://www.adjust.com/glossary/active-user/>.
- [4] The 7 Biggest Social Media Sites in 2020. Search Engine Journal. (2020). Retrieved 4 December 2020, from <https://www.searchenginejournal.com/social-media/biggest-social-media-sites/>.
- [5] 10 Kinds of Advertising. Bizfluent. (2020). Retrieved 4 December 2020, from <https://bizfluent.com/info-7736409-10-kinds-of-advertising.html>.
- [6] The Now: What is Targeted Advertising?. GCFGlobal.org. (2020). Retrieved 4 December 2020, from <https://edu.gcfglobal.org/en/thenow/what-is-targeted-advertising/1/>.
- [7] Passive vs Active Advertising: Which One Is Right for Your Brand. Instapage.com. (2020). Retrieved 4 December 2020, from <https://instapage.com/blog/passive-vs-active-advertising>.
- [8] What is Instagram? - Definition from WhatIs.com. SearchCIO. (2020). Retrieved 26 December 2020, from <https://searchcio.techtarget.com/definition/Instagram>.
- [9] Inglis, A. (2020). 8 Massive Benefits of Using YouTube For Business. Grow. Retrieved 26 December 2020, from <https://wearegrow.com/8-massive-benefits-of-using-youtube-for-business/>.
- [10] Explained: What is YouTube?. Webwise.ie. (2020). Retrieved 26 December 2020, from <https://www.webwise.ie/parents/what-is-youtube/>.
- [11] Millwood, A. (2016, July 4). 5 Most Significant Instagram Benefits for Businesses. Yotpo. <https://www.yotpo.com/resources/instagram-pics-better/>
- [12] ResearchGate. 2021. EFFECTIVENESS OF ADVERTISEMENTS IN SOCIAL MEDIA.
[online] Available at: <https://www.researchgate.net/publication/330855222_EFFECTIVENESS_OF_ADVERTISEMENTS_IN_SOCIAL_MEDIA> [Accessed 20 January 2021].
- [13] Taylor & Francis. 2021. Engagement With Social Media And Social Media Advertising: The Differentiating Role Of Platform Type. [online] Available at: <<https://www.tandfonline.com/doi/full/10.1080/00913367.2017.1405754>> [Accessed 20 January 2021].
- [14] Globaljournals.org. 2021. [online] Available at: <https://globaljournals.org/GJMBR_Volume1_8/2-Social-Media-Advertising-Response.pdf> [Accessed 20 January 2021].
- [15] Wright, E., Khanfar, N., Harrington, C. and Kizer, L., 2010. The Lasting Effects Of Social Media Trends On Advertising. Journal of Business & Economics Research (JBER), 8(11).
- [16] Ford, J. (2019, December 01). What Do We Know About Social-Media Marketing? Retrieved January 20, 2021, from <http://www.journalofadvertisingresearch.com/content/59/4/383>

Investing in Products with the Greatest Demand on Online Stores during the Pandemic

¹Titan Hassya, ²Muhammad Fauzi Hanif, ³Alvian, ⁴Ford Lumban Gaol, ⁵Tokuro Matsuo

^{1,2,3,4}Information Systems Department, School of Information Systems, Bina Nusantara University, Jakarta, Indonesia 11480

⁵Computer Science Department, BINUS Graduate Program – Master of Information System, Bina Nusantara University, Jakarta, Indonesia 11480

⁶Computer Science Department, BINUS Graduate Program - Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480

⁷Advanced Institute of Industrial Technology, Tokyo, Japan.
Email: 5fgaol@binus.edu, 7matsuo@aiit.ac.jp

ABSTRACT

The year 2020 was coming, and everyone expected the year to be better than the last, or at least stays the same as how it was last year. Due to a disease named Covid-19, every aspect in our lives were affected negatively, especially when it comes to health and businesses. Our team has decided to provide a solution or at least give the readers a conclusion that might be useful on how to overcome economy crisis during the pandemic.

Keywords: Investing, Greatest Demand, Online Stores

1. INTRODUCTION

On December 31, 2019, reports from civilians regarding diseases occurring in the lungs were discovered in Wuhan City, which is located in the capital of Hubei Province, China, resulting in the Chinese health department to report of an outbreak of a new disease, called COVID-19 [2]. The disease started to spread across other locations of the country in only a matter of a few days. On the beginning of April, the epidemic began to be regulated in China by maintaining a number of methods, for example patient isolation, usage of protective suits, and medical control. Albeit these efforts, the disease quickly spread, resulting the total of COVID-19 cases to exceed 1 million and still currently increasing rapidly afterwards.

Online shopping nowadays has become a popular way of shopping for everyone all over the world [3]. This new said innovation for vendors to interact with consumers does not only brings a great number and variety of merchandise for vendors and others to potential consumers, but it also skyrockets numerous business activities and huge market as a result.

Because of the many advantages and benefits that can be obtained, people always mentioned that they prefer online shopping over the usual conventional shopping nowadays. Online shopping or marketing via internet is the use of technology via computer to be used for better marketing method [4].

Online shopping is a type of commerce via electronic devices that gives buyers and vendors the ability to directly do transactions of various kinds from a vendor over the Internet a website or a mobile application. Consumers begins the transaction by finding a product that interests them by opening the website of the company directly or by searching among third-party vendors or dropshippers using the search engine, then it shows the product's current stock and pricing at various vendors[5]. Starting from 2020, customers will be able to shop online by utilizing a wide range of unique gadgets and devices, such as smartphones, Smart TVs, and desktop computers[6, 17].

During the pandemic, people have been using online shops as a way for them to be able to purchase their needs without having to go outside[7, 18]

II. LITERATURE REVIEW

Online shops have been a main of discussion recently, and during the Covid-19 pandemic in 2020, discussions and talks mentioning about online shops has skyrocketed. A report about Covid 19 and e-commerce by Netcomm Suisse Observatory concluded that consumers tend to focus purchases more on essential products, shop more often online, and prefer to browse as well as spend more time on digital store and entertainment sites[8, 19].

1. Covid-19 and E-commerce by UNCTAD

Based on the following summary from the documentation stated that participants across every country shows signs which resulting in lowering their average expenditures per online transaction for most of the categories [9, 20]. Categories such as media, food & beverage, as well as gardening represent the only categories with a visible increase. The decline rates are visible for categories that involves recreation, such as travel tourism, consumer electronics, and fashion gives us a great example on why the pandemic has affected many parts of economy, though some of them are not impacted in a good way.

and risk getting exposed to the toxic air. In this research, we would like to analyze how many people are beginning to rely on online shopping and start to utilize it as

part of their everyday life. In this research we would like to find out about the data we analyze later and give the readers the solutions they might find useful [10, 21].

We worked on this research to help any main players in online shopping, such as marketers, directors, etc. or even customers to guide them on how to utilize their role in online shops effectively; by that we mean that every role has its own tasks to do and every single thing matters for the sake of any department that matters in this research [11].

Article will be helpful to further analyze consumer's demand thus further improving the system applied in online shop applications/websites.

2 E-commerce trends during COVID-19 Pandemic

The literature concludes that they have reviewed and investigating the effects and risks of the currently ongoing outbreak. The literature states that the authors are interested in how coronavirus spreads worldwide and how it affects the e-commerce in detail of not only in China rather how it effects globally. Making other people aware about this problem will be able to provide readers better information in people and knowledge on how e-commerce, business, and economies of countries are being hurt by the pandemic [12]. How it impacts the e-commerce will encourage other researchers to investigate more deeply and thoroughly in this area such as e-commerce trends on how they are changed by the pandemic and trends in the upcoming generation [13].

Our research will be only focused on which category product has the highest demand for investors and data analyst to study from, but our research will not only be limited to that aspect, by that it doesn't mean other categories outside of online shops can't utilize this research.

III. RESEARCH DESIGN AND METHODS

Since our research theoretically has an unspecified scope which is done on purpose to analyze and obtain the general conclusion, this research requires us to extract data from consumers of various regions across the world using online quantitative research from countries all across the world. Please be aware that this survey will also be performed during the Covid-19 period, all kinds of face-to-face interviews will not be possible. The perfect methods of gaining an accurate data is by obtaining data from trusted secondary sources, and we have decided to choose our source of data from UNCTAD [14].

UNCTAD is the part of the United Nations that manages and responsible with various departments involved with business such as trade, investment, and problems around economy improvement[15]. Their goals are to boost trade, investment and others to be taken from assets of developing countries and assist them in their efforts to integrate into the world economy. UNCTAD has conducted the following survey partnering with several companies, Netcomm Suisse eCommerce Association, Inveon, furthermore with the Brazilian Network Information Center (NIC.br) [16]

The countries the survey conducted on were Brazil, China, Germany, Italy, the Republic of Korea, the Russian Federation, South Africa, Switzerland, Indonesia and Turkey.

As for our secondary sources, our choice will be the paper from Indonesia Data.

Our main questions are mostly related to what consumers prefer to purchase nowadays, but not limited to time where Covid 19 started to strike around March 2020. Our sample will be taken in Indonesia, where the data was conducted from.



Figure 1. Active online Shoppers during Covid 19 [1]

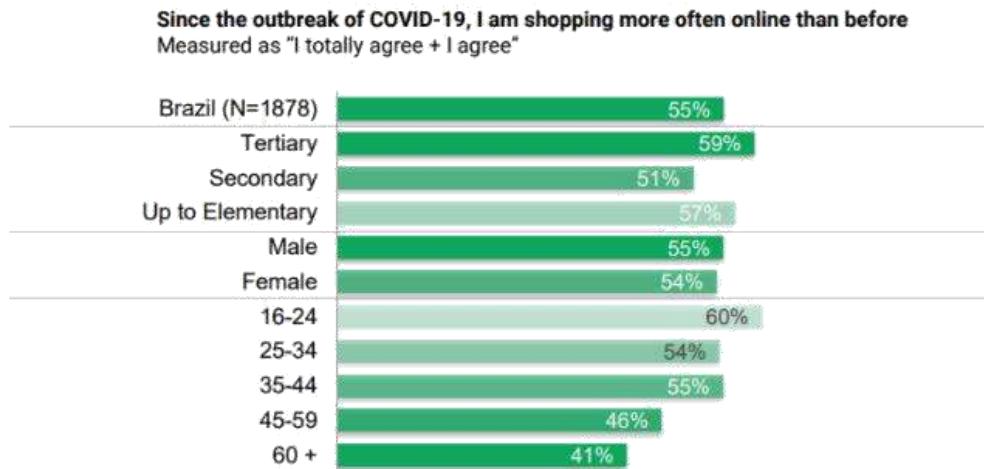


Figure 2. The result of the tendency of shopping on the Covid 19 outbreak [14].

The graphs above show the percentage of types of goods that are bought by online shoppers as well as their age. We can see that food and beverages have high demands during the pandemic, but it doesn't really mean that this category is the best choice for investors to invest in per se. There are other categories such as house hold products, cosmetics, and electronics that don't seem too far off when it comes to demand.

Table 1. The top ten websites that most online websites used [16]

Sr.no	Retail website	Millions
1	Amazon.com	4059M
2	Ebay.com	1227M
3	Rakuten.co.jp	804M
4	Samsung.com	648M
5	Walmart.com	614M
6	Appel.com	562M
7	Aliexpress.com	532M
8	Etsy.com	395M
9	Homedeport.com	292M
10	Allegro.pl	272M

(Andrienko, 2020)

The following table shows top ten of most online websites used in descending order provided with the number of users. The table above doesn't necessarily give a solution to which good category is best to invest in, but rather as a piece of information that can be used. Information analysts make recommendations on how the organization should go forward and forecasts about future consumer demands.

CONCLUSION

COVID-19 has made quite a significant negative impact on business of the world, although mostly it impacts the world in a negative way, there are a few cases that actually impacts positively. Overall, e-commerce is starting to skyrocket due of the pandemic. Coronavirus has forced consumers and sellers to use any kinds of interaction between each other via internet and treat it as a habit as their daily routine. Besides that, many problems will be faced by vendors in e-commerce, longer delivery time, and social distance and lockdown.

E-commerce players seeing a big rise in volume will be ready, and must be ready to do any kinds of obstacles in their power to keep heir new customers and visitors in check various interactions such as through promotions and expansion of the product range. The competition intensifies with consumers now are able to utilize price engines and referral sites as a feature within the store to find the best deals online.

IMPLICATIONS AND CONTRIBUTIONS TO KNOWLEDGE

Our proposed project is very essential for any types of readers, especially the ones who are looking to invest on selling goods in online shops. This, however, is available for non-investor such as those who play a role in an online shop (or those who is planning to) to adjust their view on consumers, since they are mostly not relying on the conventional methods anymore.

Readers can also utilize this research to further improve their performance and gain a significant advantage overall in understanding what their clients and consumers want to have their hands on, and providing the most effective user-friendly approach.

There are several practical implications that readers can obtain in this research, such as improving conventional methods of shopping to online shopping, or prioritizing the reader's funds or main attention to another category that may seem fit.

For theoretical implications, as we have explained in the previous section, our research may be of use

So far, this is greatest shift of people migrating to online shopping popped out among consumers and retailers in emerging economies, especially in developing countries, based on the survey.

Transactions made by consumers with the most product categories has increased ranging from six to 10 percent, information and communications technology, pharmaceuticals, education, household product and cosmetics with the biggest gains so far.

Being aware of the conclusions in mind, our paper has provided inevitable facts that are not limited to where the data was gained from, The paper is proven that users now are more likely to order food and beverages on online stores due to the currently ongoing epidemic, it is advised for the government to maintain and regulation to prioritize to assure the hygiene of food that are being sold online.

for the reader's assumption on how greatly the Covid-19 pandemic has affected our lives.

This study shows the general effects and the silver lining on how successful online shops at the country level, initiating on either private or public cooperation, have been vital on reacting quickly to the pandemic catastrophe and restrict its negative impacts. It investigates the protocol measures that the private sector has globally made the most influence to support COVID-19 restoration plans.

References

- [1]UNCTAD (2020). Covid-19 and E-Commerce Findings from a Survey of Online Consumers in 9 Countries, 21. https://unctad.org/system/files/official-document/dtlstictinf2020d1_en.pdf
- [2] Adika, R., & Subandrio, S. (2021). The Effect of Electronic Commerce and Brand Awareness on Purchasing Decisions at Shopee Online Shopping. BIMA JOURNAL: Business, Management, and Accounting Journal, 2(1), 53-66. <https://doi.org/10.37638/bima.2.1.53-66>
- [3] Afrianto, A. P., & Irwansyah, I. (2021). Exploration of Community Conditions in Choosing Online Shopping through Shopee During the Covic-19 Pandemic in Indonesia [Eksplorasi Kondisi Masyarakat dalam Memilih Belanja Online melalui Shopee Selama masa Pandemi Covid-19 di Indonesia]. Jurnal Teknologi Dan Sistem Informasi Bisnis, 3(1), 10-29. <https://doi.org/10.47233/jteksis.v3i1.181>
- [4] Agustini, P. (2021). Netizens Increase, Indonesia Needs to Increase Cultural Values on the Internet [Warganet Meningkat, Indonesia Perlu Tingkatkan Nilai Budaya di Internet]. Kominfo. <https://aptika.kominfo.go.id/2021/09/warganet-meningkat-indonesia-perlu-tingkatkan-nilai-budaya-di-internet/>.

- [5] Amrullah, M. F. (2021). The Effect of Celebrity Endorser, Brand Image, and Electronic Word of Mouth on Purchases of Shopee E-Commerce Users in Indonesia [Pengaruh Celebrity Endorser, Brand Image, dan Elektronik Word of Mouth terhadap Pembelian Pada Pengguna E-Commerce Shopee di Indonesia]. Journal of Economics, Business, & Entrepreneurship, 2(1), 1-5.
<https://doi.org/10.29303/alexandria.v2i1.28>
- [7] Batubara, B. S., Rini, E. S., & Lubis, A. N. (2021). Effect of Customer Trust, Tagline, Flash Sale, and Ease of Use on Purchasing Decisions (Case Study on Shopee Marketplace Users in Medan City). International Journal of Research, 8(2), 107–112. <https://doi.org/10.52403/ijrr.20210218>
- [8] Baubonienė, Ž., & Gulevičiūtė, G. (2015). E-Commerce Factors Influencing Customers' Online Shopping Decision. Socialnės Technologijos, 5(1), 62–73.
<https://doi.org/10.13165/ST-15-5-1-06>
- [9] hatti, A., Rehman, S. U., Kamal, A. Z., & Akram, H. (2021). Factors Effecting Online Shopping Behaviour with Trust as Moderation. Jurnal Pengurusan, 60, 109-122. <https://doi.org/10.17576/pengurusan-2020-60-09>
- [10] Ching, K. C., Hasan, Z. R. A., & Hasan, N. A. (2021). Factors Influencing Customers in Using Shopee for Online Purchase Intention in East Coast Malaysia. UMT Journal of Undergraduate Research, 3(1).
<https://doi.org/10.46754/umtjur.2021.01.006>
- [11] Cho, Y. C., & Sagynov, E. (2015). Exploring factors that affect usefulness, ease of use, trust, and purchase intention in the online environment. International Journal of Management & Information Systems, 19(1), 21-36.
<https://doi.org/10.19030/ijmis.v19i1.9086>
- [12] Gupta, P. (2011). Shopping Impulses, Online vs Off. The New York Times.
<https://www.nytimes.com/roomfordebate/2011/12/01/save-america-shop-at-work/shopping-impulses-online-vs-off>. Accessed on 23 October 2021.
- [13] Handayani, N. T., & Usman, O. (2021). The Effect of Online Customer Review, Influencer Marketing, Quality Website on Purchase Decisions Online on Online Marketplace Shopee. SSRN. <http://dx.doi.org/10.2139/ssrn.3768483>
- [14] Hermawan, D. J. (2021). Factors Affecting Interest in Buying Online [Faktor-Faktor Yang Mempengaruhi Minat Beli Online]. Jurnal Ilmiah Ecobuss, 9(2), 100-110. <https://doi.org/10.51747/ecobuss.v9i2.848>
- [15] Hertanto, A. D., & Sulhaini, H. L. E. (2020). Effect of Flash Sale Method, Product Knowledge and in Home Shopping Tendency Toward Customer Online Purchase Decisions. RJOAS, 6(102), 2020-06.
<https://doi.org/10.18551/rjoas.2020-06.12>
- [16] Huseynov, F., & Yıldırım, S. Ö. (2016). Internet users' attitudes toward business-to-customer online shopping: A survey. Information Development, 32(3), 452-465. <https://doi.org/10.1177/0266666914554812>

- [17] Husti, I., & Mahyarni, M. (2019). Islamic leadership, innovation, competitive advantages, and performance of SMEs in Indonesia. *East Asia*, 36(4), 369-383.
<https://doi.org/10.1007/s12140-019-09325-7>
- [18] iPrice. (2021). Explore the Competition of Online Stores in Indonesia (Telusuri Persaingan Toko Online di Indonesia). iPrice Group Sdn Bhd.
<https://iprice.co.id/insights/mapofecommerce/>
- [19] Javadi, M. H. M., Dolatabadi, H. R., Nourbakhsh, M., Poursaeedi, A., & Asadollahi, A. R. (2012). An analysis of factors affecting on online shopping behavior of customers. *International Journal of Marketing Studies*, 4(5), 81.
<https://doi.org/10.5539/ijms.v4n5p81>
- [20] Javed, S. A., & Javed, S. (2015). The impact of product's packaging color on customers' buying preferences under time pressure. *Marketing and Branding Research*, 2(1), 4-14. <https://doi.org/10.33844/mbr.2015.60293>
- [21] Neger, M., & Uddin, B. (2020). Factors Affecting Consumers' Internet Shopping Behavior During the COVID-19 Pandemic: Evidence From Bangladesh. *Chinese Business Review*, 19(3). 91 - 104. DOI: <https://doi.org/10.17265/1537-1506/2020.03.003>

Bhatti, Anam., Akram, Hamza (2020). E-commerce trends during COVID-19 Pandemic. https://www.researchgate.net/publication/342736799_E-commerce_trends_during_COVID-19_Pandemic

The Adoption of Artificial Intelligence Technology in Parking Security System

¹muhammad Rifqi Alhafizh, ²albertus Baskara Yunandito Adriawan, ³darryl Fernaldy, ⁴Ford Lumban Gaol, ⁵Tokuro Matsuo

^{1,2,3}Information Systems Department, School of Information Systems, Bina Nusantara University, Jakarta, Indonesia 11480

⁴Computer Science Department, BINUS Graduate Program - Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480

⁵Advanced Institute of Industrial Technology, Tokyo, Japan

Email: ¹muhammad.alhafizh@binus.ac.id, ²albertus.adriawan@binus.ac.id,

³darryl.fernaldy@binus.ac.id , ⁴fgaol@binus.edu, ⁵matsuo@aiit.ac.jp

ABSTRACT

Artificial intelligence (AI) refers to human-like intelligence exhibited by computers, robots, or other machines. In popular use, artificial intelligence refers to the ability of a computer or machine to mimic the ability of the human mind to learn from examples and experiences, recognize objects, understand and respond to language, make decisions, solve problems, and combine these and other abilities to perform functions that humans might perform. Artificial Intelligence requires experience and data to smarten up the technology. The most important things in making Artificial Intelligence are learning, reasoning, and self-correction. At the learning stage, AI gives machines the ability to learn tasks without requiring a defined programming language. Then at the reasoning stage is the stage where the reason AI is applied in a technology. At the self-correction stage is the stage where the AI is refining itself from and learning from experience in order to minimize errors or problems that exist.

Then, this matter is concerned with the parking lot to be discussed. Parking can be interpreted as public facilities available in agencies or offices that serve to store vehicles. Vehicles entering the parking area become tens or even thousands, because it requires a parking system and management area. Such arrangements are capable of parking procedures and even other support systems such as adequate parking facilities and infrastructure, another function is to create and develop parking systems in general to provide safety and comfort.

The methodology obtained from this problem is, check the research model to check and find out the comparison of vehicle users Parking System Security between cars or motorcycles. As well as the usual parking lots visited. Then, we created a questionnaire. This study used questionnaires with the aim to get respondents' results about the Parking Security System. Our sampling method is based on respondents aged 17 years to 20 years. Questionnaires are submitted through the Google Form.

Keywords: Artificial Intelligence; Parking Security System; Information Systems

1. INTRODUCTION

In this modern era, most people have their own parking systems and facilities. Parking lots usually use the services of vendors or providers of parking facilities and still use paper as a receipt for parking receipts. Today's parking systems still have their own drawbacks, although they use a high-level ticketing system, and cameras, they are still lacking in data collection of the number of vehicles entering due to manual techniques in data collection[1]. The solution to the problem can be done by creating a system capable of recording a large number of vehicles entering when entering the parking area[2].

This system can be applied using image recognition technology and OCR algorithm (Optical Character Recognition Technique), data from vehicle plate images converted into text or numbers and can be stored in a database, data from

vehicle plates that have been stored and then matched with photos of vehicles, with the help of the system can be integrated with the camera so that the system can know the duration of its parking [3]. As noted in [4] this technology can improve the safety system by reading the vehicle's license plate. Parking Security Systems that use Artificial Intelligence (AI) have a unique way of working. This technology can reduce excessive paper usage and can reduce congestion to get into the parking lot. Research question is a clear, focused, short, complex question that plays a major role in a research[5]. There are 7 research questions that we ask respondents. The first is that we ask about the vehicles used by respondents [8]. The second is that we ask how often respondents use vehicles. Thirdly we asked how often respondents parked their vehicles in public places[9]. The fourth is that we asked for a long time to queue while picking up a parking ticket. Fifth, we asked for efficiency when queuing while picking up parking tickets. The sixth question is that we ask respondents what they think about the waste of paper on parking tickets. And the last question we asked was the opinion in implementing artificial intelligence (AI) systems to replace the current parking method [10].

2. LITERATURE REVIEW

2.1 Artificial Intelligence (AI)

Artificial intelligence (AI) refers to human-like intelligence exhibited by computers, robots, or other machines. As noted in [11] In popular use, artificial intelligence refers to the ability of a computer or machine to mimic the ability of the human mind to learn from examples and experiences, recognize objects, understand and respond to language, make decisions, solve problems, and combine these and other abilities to perform functions that humans might perform.

As seen in [12] AI can analyze more and deeper data using neural networks that have many hidden layers. Building a fraud detection system with five hidden layers that was almost impossible for a few years ago. With the power of computers and big data, everything can be done easily.

AI can add intelligence to existing products. Typically, products commonly used by users can be improved using AI capabilities, such as siri examples that are added as features in Apple products. Smart machines can be combined with big data to improve many technologies at home and at work, from security intelligence to investment analysis[13].

2.1 Optical Character Recognition (OCR)

Optical character recognition or optical character reader (OCR) is the electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo (for example the text on signs and billboards in a landscape photo) or from subtitle text superimposed on an image (for example: from a television broadcast). As seen in The way OCR works, namely leveling the image according to its slope, analyzing the part of the image analyzed in accordance with the software. OCR automatically identifies and adjusts the direction of the image. Then, OCR can separate each character and letter[14].

2.2 Parking Security System

Parking spaces in the current era already have advances in technology. As seen in [15] Almost all parking spaces at the time of entry no longer require humans to provide proof of vehicle entry, the technology used only uses photos of license plates and proof of vehicle entry will appear after the driver presses the button on the engine. This makes it easy for parking lots not to have to write license plates and hours of entry manually, but it makes paper wasted for free and still creates congestion. By applying AI to the parking lot, the driver just goes into the parking lot and then the license plate will be in the photo with AI and the data and photos will be saved automatically. This will reduce the use of paper and can reduce congestion that occurs in search of parking spaces[15].

2.3 Image Processing

Image-processing are mathematical or statistical algorithms that change the visual appearance or geometric properties of an image or transform it into another type of dataset. Image processing and analysis often require fixed sequences of local operations to be performed at each pixel of an image. As seen in [16].

2.4 Vendor

Vendors are parties that provide raw materials, services, and products resold by other companies. Vendors themselves are individually conducted companies. The resulting product will be sold to the end consumer. Then, vendors not only sell raw materials, but vendors also sell semi-finished products or components to make a finished product. Vendors are very important for the smooth operation of a business. Vendors are also sometimes used in events such as concerts, or weddings. Vendors also have such duties and responsibilities, meeting every demand from consumers. Then, vendors make sure every product sold to consumers is a quality product and worth using. Also, vendors make sure the delivery of goods / services is in accordance with the deadline. Lastly, vendors are obliged to provide the best service to their customers.

2.5 Database

The definition of a database itself is a collection of data stored systematically on a computer operated using software or applications that can generate information. This database includes several specifications, such as data type, data structure, and also restrictions on the data to be stored. Databases have databases that are very important aspects because they serve as processed data warehouses. The database itself organizes data, delivers duplication of data, avoids relationships between unclear data and also somewhat complicated updates.

The process of entering and retrieving data from a database requires software such as database management systems or commonly said (DBMS). DBMS is a software system that allows users to control, access data practically and efficiently. There are several DBMS functions that can be handled such as data definition, handling user requests to access data, checking data security and integrity defined by DBA (Database Administrator), handling failures in accessing data caused by system and storage media (disk) and also handling the performance of all functions efficiently. The main purpose of DBMS is to provide an abstract review of the data to the user. So that the system can hide information about how it can be stored, maintained, and accessed efficiently.

2.6 Information Systems

Information system is a formal, sociotechnical, organizational system designed to collect, process, store, and distribute information. In a sociotechnical perspective, information systems are composed of four components: task, people, structure (or roles), and technology. As noted in [4] Information systems can be defined as an integration of components for collection, storage and processing of data of which the data is used to provide information, contribute to knowledge as well as digital products.

As seen in [14] Information systems can be viewed as having five major components: hardware, software, data, people, and processes. The first three are technology. These are probably what you thought of when defining information systems. The last two components, people and processes, separate the idea of information systems from more technical fields, such as computer science.

The source of the information system is data. Data is a reality that describes an event and real unity. Data is a form that is still raw so it needs to be processed further. Data is processed through a method to produce information, data can be in the form of symbols such as letters, numbers, sound forms, images, and so on.

3. METHODOLOGY

3.1 Research Methods

This journal examines research models to examine and to find out the comparison of Parking System Security user vehicles between cars or motorcycles. As well as the usual parking lots visited. (Table 1)

Table 1: Comparison of Parking Security.

No.	Vehicle Types	Number of Vehicle parked	
		Week 1	Week 2
1	Car	4000	5994
2	Motorcycle	6000	5999
3	Truck	300	350
4	Bus	100	150

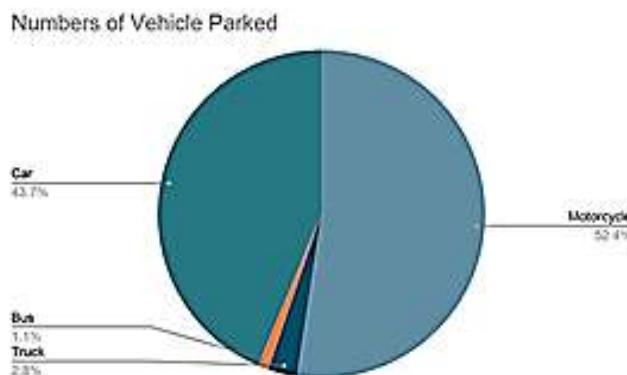


Figure 1: Comparison Chart of Vehicle Parked

Based on the data from the chart on Figure 1, vehicles parked in the parking lots are trucks (2.8 percent), buses (1.1 percent), cars (43.7 percent), and the most parked vehicle is motorcycles (52.4 percent). From the Table 1 can be shown that, the result of the answer is, more motorcycle users in Indonesia. The detailed information can be seen in table 1.

3.2 Data Gathering Methods

This study used questionnaires with the aim to get respondents' results about the Parking Security System. Our sampling method is based on respondents aged 17 years to 20 years. Questionnaires are submitted through the Google Form as shown on Figure 3.

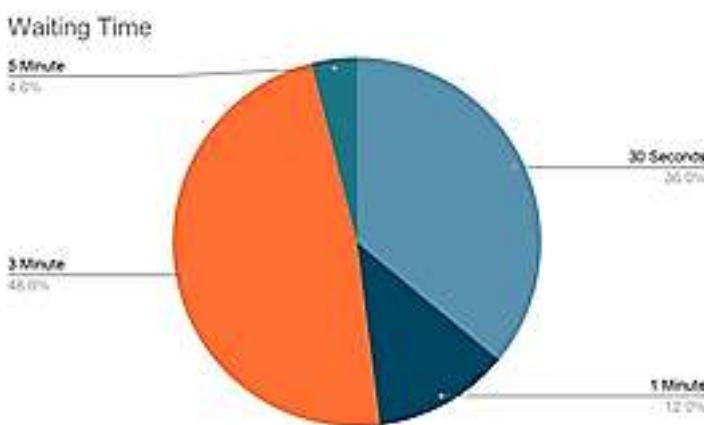


Figure 2: Comparison Chart of Respondents' Waiting Time

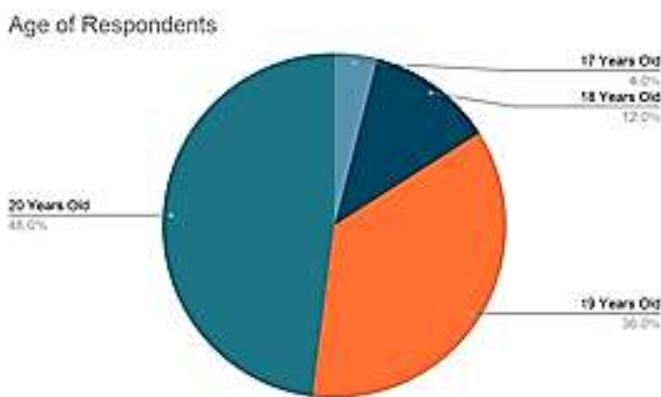


Figure 3. Comparison Chart of Respondents' Age

Table 2: List of Artificial Intelligence's Role in Changing Parking Ticket

No.	Age	Waiting Time	Vehicle	Number of Respondent
1	19	30 Seconds	Motorcycle	9
2	20	3 Minutes	Motorcycle	12
3	18	1 Minute	Public Transportation	3
4	17	5 Minutes	Car	1
Total				25

Based on the data that obtained from the questionnaires, the respondents obtained are 17 years old as many as 1 respondent (4 percent), 18 years old as many as 3 respondents (12 percent), 19 years old as many as 9 respondents (36 percent), and the 20 years old as many as 12 respondents (48 percent) as shown on Table 2. Beside the age of the respondents, there is also the respondents' waiting time as shown on Figure 2. The data obtained are 9 respondents who wait for 30 seconds (36 percent), 3 respondents who wait for 1 minute (12 percent), 12 respondents who wait for 3

minutes (48 percent), and 1 respondent who waits for 5 minutes (4 percent). The detailed information can be seen in table 2.

3.3 Data Analysis Methods

There are two methods in analyzing a data, namely qualitative data and quantitative data. Qualitative data method is data in research that explains a phenomenon based on things that generally cannot be calculated. Therefore, this data is called qualitative data because it is based on the quality of an object or phenomenon. This data is explained in the form of numbers and statistics, qualitative data is generally presented using descriptive explanations. In addition to qualitative data, there is also quantitative data which is a type of data in research that can be measured, calculated, and can be described using numbers. Generally, data like this are used to explain clear phenomena and there are already measuring instruments. Quantitative data is usually obtained when conducting statistical research. This research is like collecting a lot of data that will then be analyzed using statistical analysis to interpret the data into a statistic.

The data we get when analyzing Artificial Intelligence Technology in Parking Security System is qualitative data. Because, from the research results and from the answers of the respondents we asked about this technology we get data in the form of numbers. Like the data from the comparison of vehicles that are usually parked in public places. Then secondly, we get comparison data from the number of respondents waiting while waiting in line to pick up a parking ticket. Furthermore, we get comparison data from our respondents who have answered questions in the form of questionnaires that we provide, such data is age comparison. Another thing, why we mention that our data is qualitative data is because there are too many motorcycle users in Jakarta, Indonesia. The number of motorcycle users ranges from more than 6000 units.

4. RESEARCH RESULTS

The results of the overall questionnaire from the title Artificial Intelligence Technology in Parking Security System Based on Information System that has been distributed to 25 respondents. Sampling method is done through Google Form from 2 to 4 December 2020. The results of the questionnaire found that 66.6% were 19 years old, 28.6% were 20 years old, and 14.3% were 18 years old as shown on Figure 4.

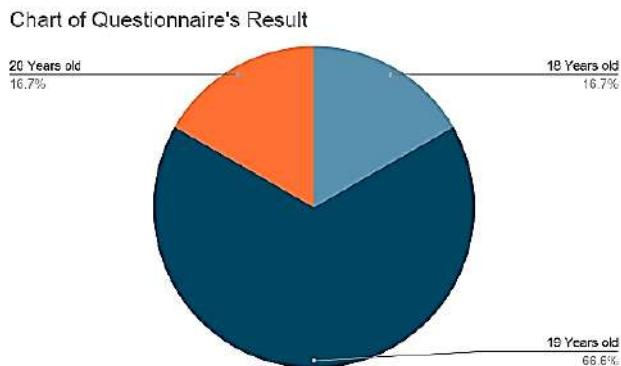


Figure 4. Results of questionnaire percentage on Artificial Intelligence Technology in Parking Security System Based on Information System

Table 3. Results of questionnaire on Artificial Intelligence Technology in Parking Security System Based on Information System

No.	Age	Vehicle	percentage of respondents
1	18	Public Transportation	14.3%
2	19	Motorcycle	66.6%
3	20	Motorcycle	14.3%

Next, we get data on how long it takes when the driver is queuing up while picking up a parking ticket. The result showed that 42.9% chose more than 5 minutes, and 28.6% chose 1 minute at queuing up while picking up a parking ticket. This can be seen in Table 3 in Data Gathering Methods. Then, the comparisons of motorcycle and car users also show that most of the respondents used car vehicles more often than the motorcycle. From this result, it can be concluded that respondents who use car vehicles queue longer to take parking tickets and according to the respondents it is very inefficient and a waste of their time.

The results of the respondents confirmed that there was a problem of waste of time as well as the use of parking ticket paper. Many respondents responded that excessive paper consumption is very wasteful and damaging to the environment. By implementing the artificial intelligence or AI system that is useful to replace ticket tickets and can save driver time. The implementation will have a positive impact on vehicle users who often park in public places.

5. DISCUSSION

The opinion we got from the results of the questionnaire given to respondents is statistical data calculations show that the average result of the comparison between parking security is motorcycle more users in the first week than in the second week. Then for car vehicles, the comparison is more in the second week than the first week. Then for truck users, the comparison is more in the second week than the first week. And lastly for bus vehicle types, the comparison is more in the second week than the first week.

In the second table there is an average lifespan of 17 to 20 years, and respondents use motorcycles, cars, and public transportation. The results of statistical data calculation showed that on average from the list of Artificial

Intelligence (AI) Role in Changing Parking Tickets, the number of respondents showed that the total of all respondents there were 25 respondents. It can be explained that the first motorcycle user who spent 30 seconds waiting amounted to 9 respondents. Then for the second motorcycle user who spent 3 minutes waiting for 12 respondents. Furthermore, public transportation users need a waiting time of 1 minute and the calculation is obtained from 3 respondents. For the latter, the results of respondents who are 17 years old require a waiting time of 5 minutes. Of the total respondents, motorcycle users outnumber public transportation users and 12 cars.

6. CONCLUSION

Artificial intelligence (AI) refers to human-like intelligence exhibited by computers, robots, or other machines. At the learning stage, AI gives machines the ability to learn tasks without requiring a defined programming language. Then, this matter is concerned with the parking lot to be discussed. Parking can be interpreted as public facilities available in agencies or offices that serve to store vehicles. Today's parking systems still have their own drawbacks, although they use a high-level ticketing system, and cameras, they are still lacking in data collection of the number of vehicles entering due to manual techniques in data collection. The solution to the problem can be done by creating a system capable of recording a large number of vehicles entering when entering the parking area. This system can be applied using image recognition technology and OCR algorithm (Optical Character Recognition Technique), data from vehicle plate images converted into text or numbers and can be stored in a database, data from vehicle plates that have been stored and then matched with photos of vehicles, with the help of the system can be integrated with the camera so that the system can know the duration of its parking. This technology can improve the safety system by reading the vehicle's license plate. The results of the overall questionnaire from the title Artificial Intelligence Technology in Parking Security System Based on Information System and has been distributed to 25 respondents. Then, comparisons of motorcycle and car users showed that many respondents used car vehicles more often. This can be concluded that respondents who use car vehicles longer queue to take parking tickets and according to the respondents it is very inefficient. The results of the respondents confirmed that there was a problem of waste of time as well as the use of parking ticket paper. Then, there are 7 research questions asked to the respondents. Of the 7 research questions, there are 6 questions that are optional and 1 question that asks for opinions from respondents.

REFERENCES:

- [1] Suhendri, Heryono, Heri and Wahyu, Ari Purno. 2019. *Improvement of Parking Security System using Artificial Intelligence Imaging Technology.*. Bandung : Journal of Information Technology; STMIK "AMIKBANDUNG", 2019, Vol. 1(2).
- [2] Potabuga, Ria Mentari Saidi. 2016. *Analisis Biaya Pengelolaan Parkir Kendaraan Pengunjung Pusat Perbelanjaan.* Yogyakarta : Ria Mentari Saidi Potabuga, 2016, Vol. 1.
- [3] Sfenrianto, Sfenrianto, Wijaya, Tendi and Wang, Gunawan. 2017. *Assessing the Buyer Trust and Satisfaction Factors .*Journal of Theoretical and Applied Electronic Commerce Research, 2017, Vol. 13. 0718–1876.
- [4] Valacich, J. and Schneider, C. 2010 *Information Systems Today – Managing in the Digital World.* 4, Upper Saddle River, New Jersey: Prentice-Hall.

- [5] Acharya, D., Yan, W., & Khoshelham, K. (2018). Real-time image-based parking occupancy detection using deep learning. In Research@ Locate (pp. 33-40). Amazon. (n.d.). Amazon Web Services, <https://aws.amazon.com>
- [6] Bong, D., Ting, K., and Lai, K. (2008). Integrated approach in the design of car park occupancy information system (coins). IAENG International Journal of Computer Science, 35(1), 7-14
- [7] Chen, M., & Chang, T. (2011). A parking guidance and information system based on wireless sensor network. In 2011 IEEE International Conference on Information and Automation (pp. 601-605). .
- [8] Ichihashi, H., Notsu, A., Honda, K., Katada, T., & Fujiyoshi, M. (2009). Vacant parking space detector for outdoor parking lot by using surveillance camera and FCM classifier. In 2009 IEEE International Conference on Fuzzy Systems (pp. 127-134). .
- [9] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
- [10] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).
- [11] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.
- [12] True, N. (2007). Vacant parking space detection in static images. University of California, San Diego, 17, 659-662.
- [13] Yu Y.J., Moon S.H., Sim S.J. & Park, S.H. (2020). Recognition of License Plate Number for Web Camera Input using Deep Learning Technique. The Journal of Next-generation Convergence Technology Association 4(6), 565-572.
- [14] P. S. Reddy, G. S. N. Kumar, B. Ritish, C. Saiswetha, and K. B. Abhilash, "Intelligent Parking Space Detection System Based on Image Segmentation," Int. J. Sci. Res. Dev., vol. 1, no. 6, pp. 1310- 1312, 2013.
- [15] M. M. Rashid, A. Musa, M. A. Rahman, N. Farahana, and A. Farhana, "Automatic parking management system and parking fee collection based on number plate recognition," Int. J. Mach. Learn Comput., vol. 2, no. 2, Apr. 2012.
- [16] R. F. O. Salma, and M. M. Arman, "Smart parking guidance system using 360° camera and Haar-Cascade classifier on IoT system," Int. J. Recent Technol Eng., vol. 8 , no. 2, pp. 864–872, Sep. 2019

A Self-Organizing Map with Neural Networks Classifier Technique for Face and Handwritten Signature Recognition System in DIP

Karthi V ^{1,*}, Nithyasai S ², Parthasarathy P ³ and Arun Kumar U ^{4,*}

^{1*} Assistant Professor, Department of Electrical and Electronics Engineering,
Builders Engineering College, Tamilnadu, 638108, India

^{1*} karthivme@gmail.com

² Assistant Professor, Department of Electronics and Communication Engineering,
Kathir College of Engineering, Tamilnadu, 641062, India

² nithyasaisivakumar@gmail.com

³ Assistant Professor, Department of Electronics and Communication Engineering,
CMR Institute of Technology, Bengaluru, 560037, India

³ arjunsarathii@gmail.com

^{4,*}Assistant Professor, Department of EEE, SRM Institute of Science and
Technology, Ramapuram Campus, Chennai, 600089, Tamilnadu, India

⁴ arun.udayakumarn@gmail.com

* Corresponding Author

Abstract. Automatic biometric recognition for individuals is a challenging problem, which has become increasingly popular in recent years by various purposes in different fields. Face and signature identification is the main issues, and until the date, no methods deliver a powerful solution to all circumstances. In this work presents an advanced technique is used for both handwritten signature and human face recognition. Mainly techniques are utilized in the image-based approaches like a self-organizing map with neural networks is removing the data from image concentration is using the 2D-DCT. The discrete cosine extracts method extracts the feature in face and handwritten signature based on the image color. The construct feature vectors calculate the DCT coefficients. A neural network classifier with SOM is used to categorize feature vectors based on the DCT which are used to classify the problem is present or not in the database. Both of the bio-metrics images are accomplished with the power rates of the grayscale pixels into many groups, and it also trained with the neural network classifier. The obtained result was evaluated and executed in MATLAB environment using a various image database of the face and handwritten images which consist of different dimensional images. Also, it observes the variation of efficiencies of the system for multiple numbers of deep layers and epochs to make comparison and contract along with a high recognition rate of 83.06% for every five trials. This technological advantage is both memory and speed usage, which makes high-speed performance and less computational requirement.

Keywords: Signature Recognition System, Self-Organizing Map (SOM), Neural Network, Digital Image Processing, Discrete Cosine Transform.

1 Introduction

Bio-metric security demands and its intrinsic economic and law enforcement applications have emerged as the most active field of research in recent years. In the past decade, it has been developed, such as Interaction of Human-Computer, analysis of biometric, coding on the content of videos and images, and monitoring significance has shown dramatic improvement in this area. Although the human brain is an important task, the identification of the face has been demonstrated to be very hard to follow a different type of color, skin, age, and gender. The various image attributes problem is, facial furniture, facial expressions, background, and light requirements are more complex. Represent a face image and signature verification system is shown in Figure 1. A unique method of identifying the face that this work is obtained from an concept. In its calculation, they represent the preprocessing step they were attempting to determine the relative pixels connected with the face-described features freely. This method designs a dramatic decrease in computational designations over earlier methods. Because skin color is unique in humans, research suggests that the investigation is more critical than colorful intensity. An intensification (grayscale) representation using the 2D-DCT image of the accreditation platform is the most common process.

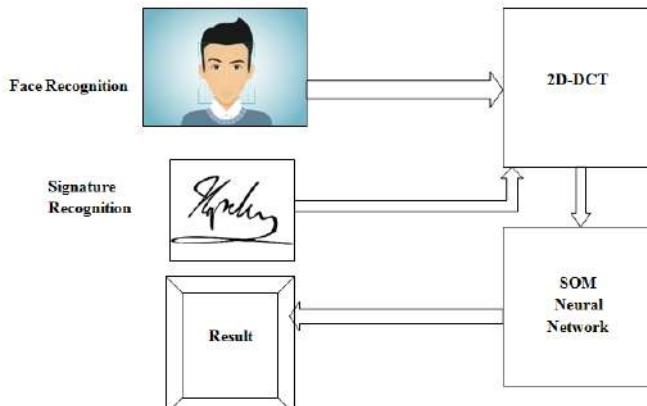


Fig. 1 Face Recognition based on Generic Representation

2 Literature Survey

Recently, single- and multi-model biometric systems research has been analysed. Below are recent multi- and unimodal biometric system studies. Skin pixel intensities are in this grayscale rendition. A face identification system approach block diagram is shown. 2D-DCT is used to evaluate all input pictures, using DCT coefficients as feature vectors and value. The self-arranging map (SOM) classifies identifier vectors as "present" or "not present" in the picture database using supervised learning. This topic is structured and shows superior picture matching in the database training system, which shows that the database image's subject is not discovered. It clarifies SOM network setup and design. The model discusses viable improvements and improves system output.

The super-resolution technique has resulted in considerable identification standard development [1]. This approach evaluates two sections for visual quality, and practise and analysis face image detection differ [2]. To identify system-helpful faces, a face image feature assessor uses two sections [3]. The discovered feature spots and their balanced geometrical patterns are compared and created. Face recognition system must consider twins, similar faces, age, same individual, etc. This article distinguishes similar twins and faces using algorithms [4].

Yawing and pitching face images may show neutral, displeasure, scream, grin, squint, and surprise emotions. This system evaluates our suggested approach utilising CMU PIE database. Our strategy improves facial recognition [5]. Face recognition accuracy depends on data complexity and extracting fundamental facial elements. LBP peculiarity space examined face image block significance [6]. The efficient face detection and identification system can recognise single and numerous face photos in a real-time database. The suggested Framework preprocesses colour pictures by removing noise and filling holes [7]. Therefore, each has distinct anatomical features. Face identification biometrics are getting increasingly prevalent [8].

This method recognises students' faces, but they may leave the classroom and the polling information can be inaccurate [9]. The best fusion level is related to the rating level, resulting in soft biometric systems [10]. This system's phase identification employs a protocol and its performance-based recognition ratio. Applied sensor level Fusion continues, the condition [11] In addition to facial photos with variable lighting and backdrops, detecting the human face from hazy noise, colour images, and photographs is difficult. Gabor Build Representation [12]. Fuzzy logic categorization is better than classes for categorising faces quickly. This study uses tool modelling to fuzzy identify sequences in detection and video displays [14]. Clarifying video faces requires multiple stages.

Approved multi-tool approaches avoid processing and reduce decision time. Diagnostic and facial classification process is two-step. First, the system colour and gigantic face are used to generate the faces [15]. Computers worry about soft processes to detect secured systems from the computer age [16]. After the big dispute, the categorization procedure has adjusted to address Internet

security and cloud service acceptance [17]. Most online methods utilise the longest ID for user/password [18]. Electronic certificates, personal identity documents, and pin codes through phone text messages are among the new user access techniques that will soon be commonplace [19].

Over the last two decades, biometric technology has challenged identification as an alternative [23-25]. They are studying Biometric Data from Biometric Science Individuals to Possible Diseases and Its Effects for Private Activities. Provide unique sufficient qualities that make some biometrics measurements an authentication system [26-28].

3 Proposed System

The proposed method can deal with the low-presentation difficulty because the Discrete Cosine Transform based self-organizing map (SOM) formation is to receive low-quality images from attribution information directly. For the exact neural network-based classification system, the operating procedures are categories into two types. At the first method is provides the complete solution exploring the total structure associated with the face and signature-less frequency domain. The resolution difference between high- and low-resolution pictures used for metric learning is recommended to be changed. These comprehensive approaches often call for well-adjusted pictures, and in order to maximise their performance, it is often required to dampen the linear disorder in real-world contexts. It reveals that the low-frequency domain local phase quantization information is mostly unaffected, making it appropriate for recognising a lower-resolution face. To begin, Local phase quantization (LPQ) discards magnitude information, despite the fact that magnitude is crucial to facial identification. For the second part, LPQ necessitates the point spread function (PSF) that makes the blur impact positive. The suggested technique relies on coarse-grained, locally-frequency-specific data. Instead of the LPQ, this explanation makes use of both level and grid data.

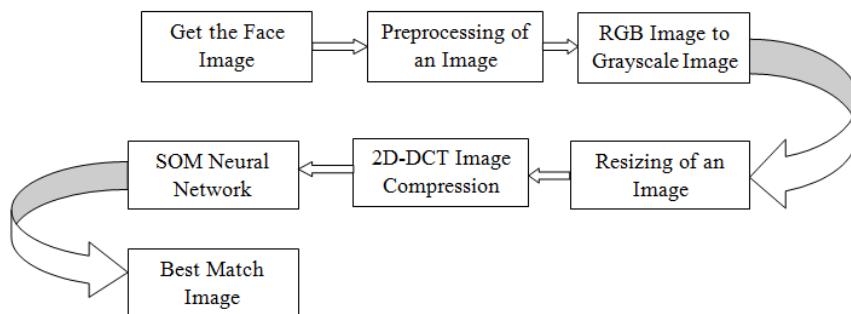


Fig 2. Proposed Methodology

Figure 2 describes the proposed Block diagram for the face and handwritten based signature analysis with the differential dataset. Mainly a self-organizing map (SOM) based Neural network accurately classifies the Trained Face images from the dataset images.

3.1 Self Organizing Feature Map and 2D DCT is a New Face Detection Technique

Faces reveal the ACE identification difficulty. This face-to-face, face-monitoring, and human-computer interaction (HCI) is crucial. Face detection works aggressively over a decade, but orientation, brightness, obstacles, and skin colour disparities complicate. This study introduces Low-based face detection. They discuss the preprocessing technique to detect face-related pixels freely in its computation. It dramatically reduces processing needs for multi-resolution picture pyramids, but these pyramids no longer require skin colour testing before a multi-system candidate prepares a list of candidate face regions. Pixel- or region-based skin region extraction is possible. Studies show that skin colour is more unique than the primary character. Because the following phase is a gradient (grayscale) representation of the segmented picture, colour space is less important. This grayscale "skin diagram" shows skin pixel intensities on a black backdrop. Skin diagram from colour input. For ease, pre-processed VT-AAST skin maps. This study uses the Database.

Our proposed technique is provided with a block diagram. Including a segmented image as an input, the first level of the region will be adjusted, and the large areas were attached to the narrow connections, the subdivision will apply to an image opening. The performance of the region labeling is done so that each section can be different from the others. In the second phase, the two-dimensional (2 D) standalone cosine is calculated for each region (DCT), and the DCT coefficients as feature vectors.

The platform uses a self-arranging map (SOM) to analyze every aspect of the vector as either "face" or "not face." The yield is having been predictable by the association faces. So, the group of locations in the original image is in the group. Ranking and evaluating this new technology, a new model is compiled for information training and testing purposes. The database containing 286 images a total of 1027 faces. Each image is available in four styles. Images are characterized not only by their original hue but also by the color of the subjects' skin. These GIF images are of size 200×125 pixels.

3.2 Handwritten Feature Recognition

An automatic signature verification system has been proposed. This work focuses on handwritten signature features and aims to synchronize and verify the signature. Signatures process data and information are collected. An image is obtained in the data sets are the sealed signature process using the signature of

scan data. Initially, two data are undergoing preprocessing operations. The features are based on the pen ending tracking features online and are based on gradient and project-based extraction where the connection is used in case the internet connection is inaccurate. Then the signature checks the system individually, and finally, their results are synchronized and checked using the signature neural classifier. The paper also compares the results of the approach to coordinating training data and information system.

3.3 Two-Dimensional discrete cosine transform

The 2DPCA method consists of the covariance matrix of definitions of the Sof N training images A_i of dimensions $m \times n$ (where $I = 1, \dots, N$) in 2Dimension. The covariance matrix S of $n \times n$ dimension is calculated by (1).

$$S = \frac{1}{N} \sum_{i=1}^N (A_i - \bar{A})^T (A_i - \bar{A}) \quad (1)$$

Where:

A is the average matrix in the training N images. The consisting set of the k eigenvectors related to the largest eigen values of the covariance matrix, $V = [V_1, V_2 \dots V_k]$, of size $n \times k$, is obtained so that the prediction of the training images about V provides the best dispersion. V is used to obtain the feature of each training image A_i . The feature vectors are obtained by(2).

$$y_{j,i} = A_i V_j \quad j = 1, 2, \dots, k; i = 1, \dots, N \quad (2)$$

In the classification proposed the similarity between the feature matrices of the images was obtained.

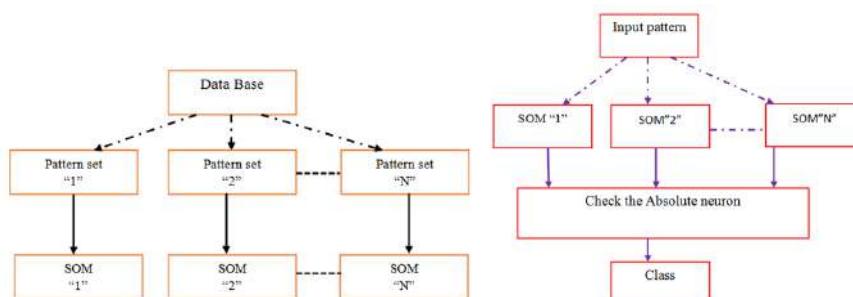


Fig. 3 SOM Training Model

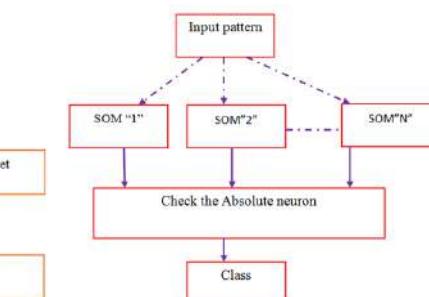


Fig. 4 Neural Network Based SOM System

When it comes to classification, the class of the test pattern is the one that is determined by the one that has the smallest distance between the test pattern and the neurons of the SOMs, as seen in Figure 4.

The most successful system recognition has been used for various problems, such as to test an image of neural networks, and the introduction of the skin color vector classification test in the presence of each DCT-based feature. Using neural networks based on several image detectors one face or face is one. Neural networks use this paperback spread because of their ability to choose simplicity and supervised type matching. A three-layer neural network design is used, with the input layer receiving either a face picture or a non-face image, the hidden layers housing single neurons in an area with a $1 \times N$ DCT coefficient vector, and the output layer having two distinct weights of 0.9 and 0.1.

Skin face colour is acquired via the DCT coefficient feature, and the neural networks are bound to carve 18×27 pixels from DC, with each volume 8×8 pixels in size. Facial features are classified as having a velocity of 0.9 after training, whereas non-facial features have a velocity of 0.1 after training (Va Ktar). A color image skin color information face detection is a very popular and useful technique. This method has the simplicity of the skin detection rules that leads to a very rapid assortment of clear benefits. The color information is used as a feature to detect from background objects, even if not entirely because the faces of human faces differ in the human face in a picture.

3.4 Self-Organizing Map with Neural Networks Classifier Algorithm

Suppose a feed-forward the neural network is whose input (or predicted variables) whose output (or response) is indicated without the loss of the vector X and any general force of the output is used to approximate a resilient function when revealing the Scalar YA regression surface (input-output function) When attempting to characterize the bond between X and Y, F_j is used. This association may be represented via the training and application of a feed-forward neural network.

The SNNC algorithm is as follows:

Step 1: Read the intensity corrected image.

Step 2: Improved versions of the associated pictures that may be used in preprocessing.

Step 3: Figure out the cutoff at each neural network iteration.

$$T = \max(r(T)) \quad (4)$$

- Where $r(T) = E(X' - X)$, where X is the image and X' is the generalized defeat identification.
- Perform directional diagram
- If convergence reached, then go to step 3, else step 2 is repeated.

Step 4: output de-noised image

Step 5: Back-multiplication of fault at the output layer, the error between the desired output S_k and Z_k output is measured by:

$$E_k = Z_k(Z_{k-1})(S_k - Z_k) \quad (5)$$

The error value calculation is generated on the hidden layer using the following formula:

$$\frac{dx}{dt} = A^T g(F_j) F_j \quad (6)$$

Step 6: Specifically, we use a correction for the fixed attachment weights between the input layer and the hidden layer.

$$DW_{ji} = nX, F_j \quad (7)$$

$$DY_o = nx, f_j \quad (8)$$

DY_o is the after removing blurring images nx is the intensities' function.

Then change connections between the input layer and the output layer by:

$$DW_{kj} = nyjE_k \quad (9)$$

N is considered to be determined empirically.

Where E_k the power energy functional static position optimization is can be written as follows

Step 7: Loop to st continuously a stop to define criterion (varied the parameters clusters).

Step 8: end

4 Results and Discussions

MATLAB 2013a is the most used manipulation and operating programme in the 2D, 3D data and picture advancing. Relevant approaches to observe and analyze simulation outcomes in image processing are implemented based on the specific operation and creative situation. This programme will provide users with the ability to manipulate and filter images in a wide variety of ways. Here the picture analysis uses precise color photos and filter for additional variation.

TESTING DATA 1: RECOGNIZED

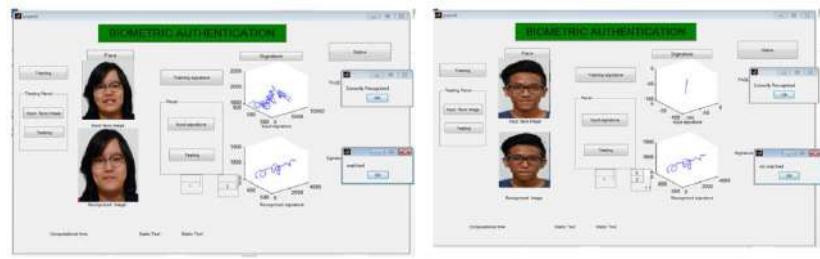


Fig. 6 Detection using SNNC

Fig.7 Detection using SNNC

Table 1 Evaluation Parameters Analysis

Performance	Value
Data Set	ORL
No of Persons	40 persons
No of Face images	40 persons of 248 images; every person has 10 face images
No of Signature	248 different signatures collected from 40 people's signatures.
Tool Used	MATLAB 2013 a
Data Set	ORL

The image recognized and not recognized signature & face verification have been shown in figure 6 and figure 7. The test takes the same meaning as input. Input image and signal begins to operate using adaptive average filter and practice using LF. Then, an algorithm based on self-organizing maps is used to do the necessary picture preprocessing. In the end, picture and non-accredited image were used to determine if SNNC could be recognized.

4.1 Analysis of Various Metrics

Table 2 Performance Comparison of Data Set 1

MEASURES	Proposed SMNN	Synthesis score fusion based SSF-MNN	LFNN	EXISTING[PCA]	EXISTING[ICA]
Precision	0.94	0.93	0.92	0.82	0.86
Recall	0.58	0.57	0.55	0.52	0.54
FMeasure	0.76	0.70	0.68	0.64	0.66
Accuracy	0.71	0.68	0.62	0.53	0.569
Sensitivity	0.72	0.57	0.55	0.52	0.54
Specificity	0.77	0.74	0.72	0.58	0.66

Correlation table 2 shows that the proposed SMNN method outperforms more traditional approaches, such as (SSF)-MNN, LFNN, PCA, and ICA, on a variety of metrics. These metrics include accuracy, recall, precision, specificity, and sensitivity. Furthermore, the false ratio is lowered to 0.08 using the suggested strategy, making our system very effective. In addition to reducing the True Positive Rate, the False Negative Rate, and the False Rejection Rate are all greatly reduced in comparison to the current ICA and PCA technique. The comparison graph is given below.

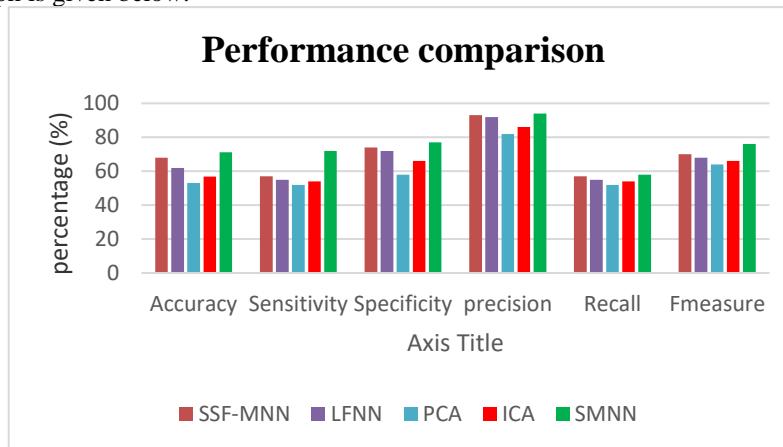


Fig. 8 Comparison graph

Figure 5 compares the suggested strategy. SMNN outperforms SSF-MNN, LFNN, PCA, ICA in sensitivity, accuracy, precision, specificity, recall, and FM. Table 2 compares multiple recognition models. Figure 6 illustrates that the proposed technique delays time less than alternative strategies.

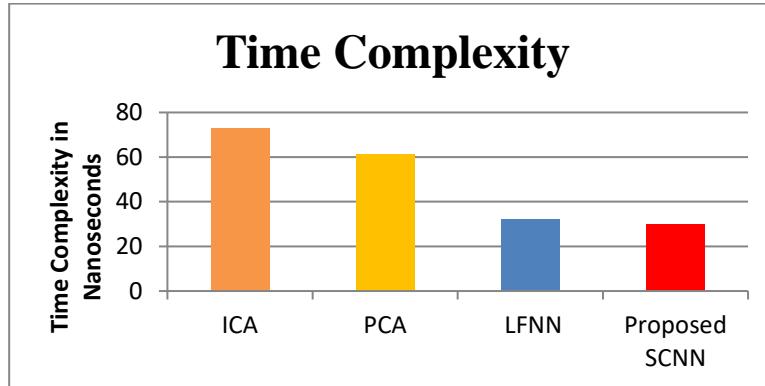


Fig. 9 Comparison of time complexity

5 Conclusion

In this system the conjunction with a SOM-based classifier, and the network identification technique that uses features derived from DCT coefficients. The MATLAB is used to evaluate the system was 25 face images database and signature analysis; it consisting of five subjects and subjected to having five models and the various facial appearances. The training about 850 eras, the organization, received an accredited rate of 83.06 % with 10 consecutive tests. The Feature Space is reduced to described above test 2; the reduced computational demands once compared to the standard DCT feature extraction methods. With this low cost, real-time hardware is much better on our system to implement. There is no commercial implementation of this strategy at present. However, it is assumed that a useful SOM-based face identification system may be possible in the future.

References

- [1] Mahood, A., Sobh, T & Elsayed, A.,(2017) “Effect Of Super Resolution On High Dimensional Features For Unsupervised Face Recognition In The Wild” in IEEE Recognition of Applied Imagery Pattern Workshop, pp. 1-5.
- [2] Lee, H.-I., S.H., & Ro, Y. M.Kim, (2015) “Face Image Assessment Lea grant with the Objective and Reactive Face Image Qualities for Improved Face Recognition” in IEEE Conference on Image processing, pp. 4021-4031.
- [3] Chen, Huang, Y.-S., S.-Y. (2015) “Ageometrical Model Based Face Identification” in IEEE conference on image processing, pp. 3106-311.

- [4] Shamugapriya, P. & Prema, R., (2017) "A Review Face Recognition Techniques for Differentiating the Faces and Twin Faces" in Conferences On Data Analytics, Energy, Communication, And Soft Computing, pp. 2899- 2902.
- [5] Madarasmi, S., Chamnongthai, K., & Petpairote, C., (2017) "A Pose and Expression Face Recognition Using Transfer Based on Single Face Neutral References" Global Wireless Summit, pp. 123-126.
- [6] Pavlovicova, J., Oravec, M., Loaderer, M., & Mazanec, J. (2015) "Face Parts Importance in the Face and Expression Recognition" in International Conferences on System, Signals and Image Processing, pp. 188-191.
- [7] Santhoshkumar, S. P., Beaulah, H. L., Alqahtani, A. S., Parthasarathy, P., & Mubarakali, A. A remote diagnosis of Parkinson's ailment using artificial intelligence based BPNN framework and cloud based storage architecture for securing data in cloud environment for the application of telecommunication technologies. Computational Intelligence.
- [8] Qasim, T., Bakhat, K. & Sammem, M.S. I., (2016) "Real Image Recognition of Human Faces" in International Conference on open Source System & Technologies, pp. 62-65.
- [9] Varol, A. & Haji, S., (2016) "Real Ti Me Face Recognition System (RTFRS)" in International Symposium on Digital Forensic and Security, pp. 107-111.
- [10] Tanriverdi, M. Samet, (2017) "Face Recognition -Based Mobile Automatic Classroom Attendances Management System" in International Conferences on Cyber words (Cw).
- [11] Chawla, S. & Arorra, S., (2015) "An Intensified Approach to Face Recognition through the Average Half Face" in IEEE India Conferences, pp. 1-6.
- [12] Kumar, A & Punyani, P., (2017) "Evaluation of Fusion at Different Levels for Face analysis" in International Conference Communication, And Automation, pp. 1052-1055
- [13] Vincent Paul, S. M., Balasubramaniam, S., Panchatcharam, P., Malarvizhi Kumar, P., & Mubarakali, A. (2022). Intelligent Framework for Prediction of Heart Disease using Deep Learning. Arabian Journal for Science and Engineering, 47(2), pp.2159-2169.
- [14] Parthasarathy, P., & Vivekanandan, S. (2020). Biocompatible TiO₂-CeO₂ nano-composite synthesis, characterization and analysis on electrochemical performance for uric acid determination. Ain Shams Engineering Journal, 11(3), pp.777-785.
- [15] Amin, M. A., Yan, H. & Poon, B., (2016) "Gabor Phase analysis on Human Face Recognition for Distorted Image" in International Conference on Wavelet Analysis and Pattern Recognition, pp. 276-281.
- [16] Abadiano, & Wait, D. A. R., (2017) "Design of classification System and Face Detection for Smart Home Security Application. International Conference on Information Technology" in Information System and Electrical Engineering, pp. 342-3347.
- [17] Fakir, M., Chabi, M. & Hatimi, H., (2016) "Face Recognition Using A Fuzzy Approach and a Multi-model System From Video Sequence" in International Conference On Computer Graphics, Visualization and Imaging, pp. 442-447.
- [18] Vijayarajeswari, R., Parthasarathy, P., Vivekanandan, S., & Basha, A. A. (2019). Classification of mammogram for early detection of breast cancer using SVM classifier and Hough transform. Measurement, 146, pp.800-805.
- [19] K. M. Rajesh and M. Naveen Kumar, "A robust method for face recognition and face emotion detection system using support vector machines," 2016 International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT), 2016, pp. 1-5, doi: 10.1109/ICEECCOT.2016.7955175.
- [20] Travieso, C.M., Dutta, M.K., Pitters-Figueroa, F.A., & Singh, A. (2017) "Biometric Identifier-Based on Hand and Hand-Written Signature classification and Contour Information" Tenth International Conference on Contemporary Computing, pp. 1-6.
- [21] U. Kumar, G. Kavya, J. Kishore and K. A. N. Raj, "BL-CSC Converter Fed BLDC Motor Drive with Sensorless Control," 2018 4th International Conference on Electrical Energy Systems (ICEES), 2018, pp. 449-453, doi: 10.1109/ICEES.2018.8443286.

- [22] Parthasarathy, P., & Vivekanandan, S. (2020). A typical IoT architecture-based regular monitoring of arthritis disease using time wrapping algorithm. International Journal of Computers and Applications, 42(3), 222-232.
- [23] Basavaraj, L. & Prathiba, M. K., (2017) "Signature Verification System Based On Wavelets" in International Conferences on Recent Advances In Electronic And Communication Technology, pp. 149-153.
- [24] Seakar, V. C. & Viswanath, (2017) "A Style Preserving Shape Representation Scheme For Handwritten Gesture Recognition" in Ninth Conference on Advances in Pattern Recognition, pp. 1-6.
- [25] Jahan, M.V. & Farimani, S. A., &Jahan, M.V. (2018) "A Hmm For Online Signature Verification Based on Movement Directions and Velocity" in Iranian Joint Congress on Fuzzy and Intelligent System, pp. 205-209.
- [26] Kumar, U.A., Ravichandran, C.S. Upgrading the Quality of Power Using TVSS Device and PFC Converter Fed SBLDC Motor. Arab J Sci Eng (2021). <https://doi.org/10.1007/s13369-021-05600-z>
- [27] Zhang, R., Ritchie, M., Li, G., & Griffiths, H. (2017) "Sparsity Based Dynamic Hand Gesture identification Using Micro -Doppler Signatures" IEEE Radar Conference, pp. 1258-1260.
- [28] Matrouk, Khaled, et al. "Energy Efficient Data transmission in Intelligent Transportation System (ITS): Millimeter (mm Wave) Based Routing Algorithm For Connected Vehicles." *Optik* (2022): 170374.

A NOVEL DYNAMIC ENERGY AMENDMENT CONTROL FOR SOLAR FED BRUSHLESS DC MOTOR FOR WATER PUMPING SYSTEM

Arun Kumar U ^{1,*}, Chandraguptamauryan K S ²,
Chandru K ³ and Rajan Babu W ⁴

^{1,*} Assistant Professor, Department of EEE, SRM Institute of Science and Technology, Ramapuram Campus, Chennai, 600089, Tamilnadu, India

^{1*} arun.udayakumarn@gmail.com

² Professor, Department of Electrical and Electronics Engineering, Guru Nanak Institutions Technical Campus, Hyderabad, 501506

² cgmauryan@gmail.com

³ Assistant Professor, Department of EEE, EASA College of Engineering and Technology, Coimbatore, 641105 Tamilnadu, India

³ chandru15387@gmail.com

⁴ Professor, Department of EEE, Sri Eshwar College of Engineering, Coimbatore, 641202, Tamilnadu, India

⁴ rajanbabu.w@sece.ac.in

* Corresponding Author

Abstract. This study optimises solar PV power for a Brushless DC motor (BLDC)-driven water pumping system utilising Dynamic Energy Amendment Control (DEAC). The improved Isolated DC-DC converter with an Adaptive Perturb & Observe (APO) algorithm-based Maximum Power Point Tracking (MPPT) technology generates steady DC power alternating current, generally by a Brushless Direct Current Motor (BLDCM) motor, to enhance solar Photovoltaic (PV). BLDCM needs solar-powered water pumping. Power conversion causes varying sizes, cost, complexity, and efficiency loss. A single-stage operating solar PV system and BLDC electric pump eliminate the isolated DC-DC converter phase to create a stand-alone solution. A BLDC motor may be controlled using a Voltage Source Inverter (VSI) to drive a solar photovoltaic array at peak power. Dynamic Energy Amendment Control (DEAC) decreases BLDC motor phase current sensing. The DEAC controller automatically regulates the inverter's PWM based on load fluctuation, therefore speed control is not needed. BLDCM speed control needs are greater because rotor position analysis requires continual self-synchronization feedback. High homeostatic functions absolute or incremental encoders are usually utilised with the BLDCM. BLDCM modelling and simulation should address motor steady-state characteristics, speed reversal, and load tolerance. The DEAC controller simulations offer accurate results under all operating situations. Test results are compared to MATLAB/Simulink results and experimentally verified under real operating circumstances.

Keywords: Solar Photovoltaic(SPV), Adaptive Perturb & Observe (APO) algorithm, MPPT, Dynamic Energy Amendment Control (DEAC), BLDC, Field Programmable Gate Array (FPGA).

1 Introduction

There has been a worldwide shift in the previous several decades with regards to energy usage, the amount of help given, and environmental concerns. Mitigation is a major issue that has to be addressed. There has been significant progress made in addressing the issue, thanks in large part to the use of renewable energy sources, electric motors with low noise pollution characteristics, and high-efficiency power transmission systems. A system of electric motor pumps is one of the most common uses of solar energy. Household uses, agricultural irrigation, and rural water consumption are just some of the various uses for solar pumping systems in outlying locations. The benefits of a water pump based on a PV system include minimal maintenance, simple installation, and dependability. The DC-DC converter employs the Adaptive Perturb & Observe (APO) algorithm to maintain consistent solar output. Figure 1 is a simplified version of the basic block diagram used to execute the stability speed control of the brushless DC motor (BLDC) and the state-of-the-art Dynamic Energy Amendment Control (DEAC).

Photovoltaic arrays power water pumps using BLDC motors. DC-DC converters employ Maximum Power Point Tracking (MPPT) technology to stabilise solar power from photovoltaic arrays and feed it to the inverter for BLDC electric pumps for agricultural application. APO-MPPT simplifies direct duty cycle control. This solution removes the APO controller and duty cycle control settings directly. Noise and duty rotation have little direct effect, thus rotation is stable and energy efficient without vibration. The greater the interference rate, the higher and more steady the system's worldwide utilisation PWM rate. Without oscillation metres, dynamic circumstances near the ideal operating point and minimal optimal monitoring performance are attained. The initial duty cycle and interrupt range gently boost the VSI DC bus voltage to start the smooth BLDC motor.

This study analyses energy's dynamic response characteristics, offering fixed/constant irradiance corrective control APO to stabilise PV power and DEAC to improve the BLDC drive system. The controller controls the APO algorithm to dynamically track BLDC drive system power input. The boost converter tracks maximum power point for this. This powers the BLDC motor's voltage source inverter. DEAC simulations operated the BLDC drive system at a constant speed at varied irradiation levels.

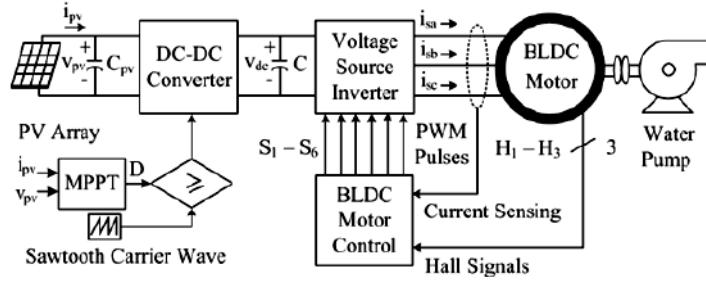


Fig. 1 Fundamental block diagram for the solar fed BLDC motor

2 Literature Survey

Due to the BLDC drive's thinning and structural benefits, household applications employ BLDC motors. It reduces computational complexity and application driver controller integrated circuit size, which is beneficial. Latest Dual Duty Modulation technique [1-2]. The BLDC motor uses a novel commutation error-based correction approach to reduce phase distortion between current and post-EMF [3]. To minimise irregular driving control and boost system dependability, an online-level pulse signal fault detection software with rapid fault tracking is needed [4]. The author suggests quick commutation error correction to increase accuracy and decrease cost. The BLDC motor control topography is first tested with no load, then gradually increased to analyse performance. The Electromotive Force (EMF) DC Connection Extended State Observer (ESO) system for voltage sources re-evaluates the new conductor. Finally, EMF redirection error is compensated [5-11].

In order to test BLDC motor drives with a single-phase power supply, one FPGA implementation employs a Single-Ended Primary-Inductor Converter (SEPIC) converter based on an induction bridge. From the three-phase VSI, the converter provides a variable DC connection voltage for the electronic commutator motor BLDC [12, 14]. A single-phase brushless DC (BLDC) motor is most suited for this task because of its high operating speed [15, 16]. The speed-independent inter-linkage flux function is preferable for sensorless BLDC motor control at lower speeds. An enhanced closed-loop system was shown based on full-speed changes [17-21].

The electromechanical control system is used to display an event-driven operating system implemented in FPGA. The supervisor provides a powerful, secure, transparent control that defines all possible directions for a Finite-State Machine (FSM) to optimize [22]. The potential stability problem of this investigation is due to various conditions of load disturbance and the ease of such control under reduced processor power. This criterion has been used to an analysis of Lyapunov stability in closed-loop systems [23]. The BLDC motor's dynamic

speed control stability is improved by implementing and testing an adaptive control technique under a variety of loads.

3 Proposed System

In this paper, we present the design and analysis of a controller for speed regulation of a BLDCM that uses corrected Dynamic Energy Amendment Control (DEAC). When operating a solar fed-BLDC motor, the user may rapidly adjust the magnetic flux and torque components thanks to the motor's direct drive torque. To get the highest possible dynamic performance from the proposed drive system, the Dynamic Energy Amendment Control (DEAC) controller is implemented.

Figure 2 shows the proposed architectural structure of a solar-powered BLDC motor. The BLDC motor operates a VSI or voltage source inverter to generate magnetic flux. It adjusts the " θ " angle of the motor to facilitate its operation and plays a key role. According to the policy of the BLDC motor, there is a supply between the two stages. EMF and the margin are formed at one point. Back-EMF has a basic sensitivity function, which then helps to detect zero cross-sections; it creates pulses with a level code. Based on the switch function in the switch table, it can be controlled. To achieve this cycle, an Advanced Dynamic Energy Amendment Control (DEAC) controller effectively stabilizes the overall operation.

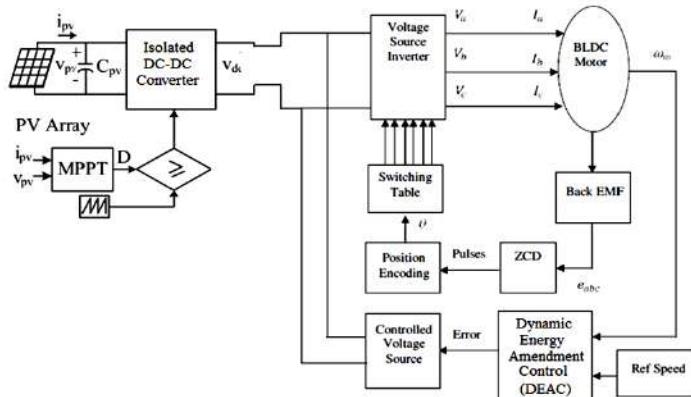


Fig 2. Proposed Methodology

3.1 Isolated DC-DC converter

Figure 3 illustrates the DC-DC converter's proposed single-switching circuit design. A High Voltage Side (HVS) circuit and a Low Voltage Side (LVS) circuit are linked together by a High Frequency (HF) transformer to produce the isolation converter. The LVS has a primary winding transformer, a storage capacitor CS, a

switch S1, and an induction L1. The secondary transformer incorporates a complete bridge rectifier and a low-frequency LC filter coupled to the diodes DS-1 through HVS-dissolved DS4. The ratio of the torque produced by the main and secondary windings of a transformer is denoted by $n = NS / NP$. In addition to its primary purpose of converting between two different formats, the converter also Changing the V1 voltage input Voltage output that is both high and stable Vdc and Solar panel status monitoring The place of greatest strength (MPP) Weak adjusting.

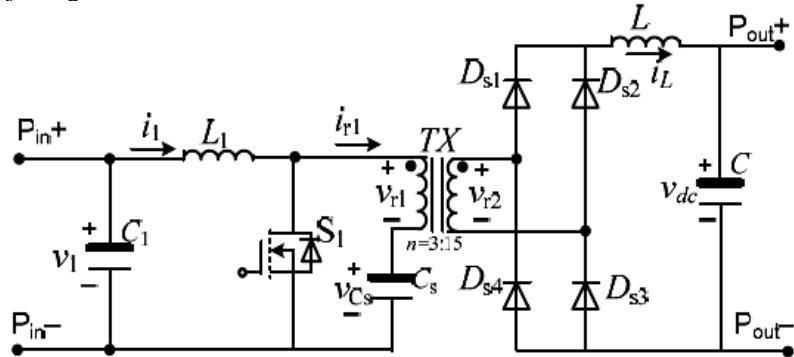


Fig 3. Proposed Isolated DC-DC converter

A single switch S1 controls the converter to simplify the operational analysis; the capacitors of all the transformers are considered to be adequate so that the DC voltage becomes low compared to the voltage surge across them. The semiconductor is also considered to be load resistant fully and a supplied converter. The converter state switch operates in two stages according to S1.

3.2 Proposed APO-MPPT Technique

Optimal performance of the combined solar/wind/fuel cell system is maintained by the use of APO-MPPT technology. For technical purposes, the power slope is flat when $DP/DV = 0$, and it is empty when the slope is positive or negative. Here is the MPPT method that uses APOs:

$$P_{pv} = P_{pv} \times i_{pv} \quad (1)$$

$$\frac{\partial P_{pv}}{\partial V_{pv}} = i_{pv} + v_{pv} \times \frac{\partial i_{pv}}{\partial v_{pv}} = 0 \quad (2)$$

$$\frac{\partial i_{pv}}{\partial v_{pv}} = -\frac{i_{pv}}{v_{pv}} \text{ at MPP} \quad (3)$$

$$\frac{\partial i_{pv}}{\partial v_{pv}} > -\frac{i_{pv}}{v_{pv}} \text{ at the left of MPP} \quad (4)$$

$$\frac{\partial i_{pv}}{\partial v_{pv}} < -\frac{i_{pv}}{v_{pv}} \text{ at the right of MPP} \quad (5)$$

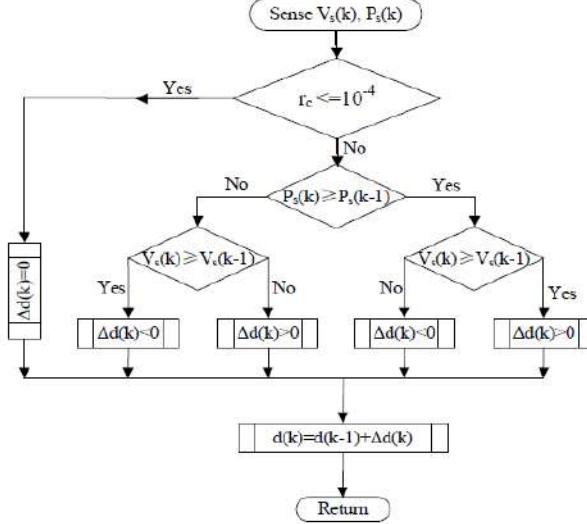


Fig 4. Flow chart for APO MPPT technique

The APO-based MPPT technique is used to improve the output efficiency of photovoltaic panels. Figure 4 shows the flow chart of the algorithm.

3.2 Optimization Steps of Proposed DEAC Controller

Step 1: The quadrature-axis stator current reference i_{qs}^* is computed from torque reference T_e^* as

$$i_{qs} = (2/3) (2/P) (L_r/L_m) (T * \frac{e}{\Psi_r}) \quad (6)$$

Where,

T =torque

L_r =rotor self-inductance

L_m =stator self-inductance

i_{qs} = axis stator current reference

Where $\Psi_r = |\Psi_r|$ lest is the evaluated value of rotor flux linkage known by,

$$\Psi_r = \left(\frac{L_{mids}}{1+\tau_{rs}} \right) \quad (7)$$

$$\text{Where, } \tau_r = \left(\frac{L_r}{R_r} \right) \text{ rotor time constant} \quad (8)$$

Step 2: The direct –axis stator current reference i_{ds}^* is gained from reference rotor flux response $|\Psi_r|^*$

$$i * ds = \frac{|\Psi_r| *}{L_m} \quad (9)$$

Step 3: The rotor flux position θ_e required for Dynamic Energy Amendment Control (DEAC) is acquired from the rotor speed ω_r and slip frequency ω_s . θ_e calculated as,

$$\theta_e = \int(\omega_r + \omega_{sl.})dt = \theta_r + \theta_{sl.} \quad (10)$$

Where,

θ_e = rotor flux position

ω_r = rotor speed

ω_s = Slip frequency

dt = differential time

Step 4: The slip frequency stator reference is estimated from the current i_{qs}^* and motor parameters

$$\omega_{sl} = \left(\frac{L_m R_r}{\Psi_r L_r} \right) i_{qs}^* \quad (11)$$

The three-phase BLDC motor current (ABC-axis) of the elements of the two-stage conversion (dq-axis) can be equal to the current elements,

$$\begin{bmatrix} I_{qs} \\ I_{ds} \end{bmatrix} = 2/3 \begin{bmatrix} \sin \theta_e \sin \left(\theta_e - \frac{2\pi}{3} \right) \\ \cos \theta_e \cos \left(\theta_e - 2\pi/3 \right) \end{bmatrix} \quad (12)$$

The three-phase current component of the i_{as} , i_{bs} , and i_{cs} , and the space within the rotation frame are completely in a blocked position. In contrast, the two-phase current components i_{ds} , i_{qs} and synchronous system are nevertheless in place to direct and quadrature axis at a constant speed.

Step 5: The FPGA's motor's internal configuration memory is then set to the reference speed.

Step 6: By applying the error and the change in the measurement error between the motor speed and the output of the reference model to the motor inverter, the suggested FPGA controller could keep the motor running at the desired speed.

4 Results and Discussions

This MATLAB2017b-based BLDC Driver system, which is based on the suggested Dynamic Energy Amendment Control (DEAC) approach, is the result of extensive research and development. The coordinated system will verify the functioning of the suggested model while the simulation of the system's function will verify the model's accuracy. This image depicts the proposed BLDCM model's implementation in MATLAB/Simulink, its usage, and its capacity to present libraries of SIM power supply system-wide setups.

Table 1. Specification of the proposed solar fed BLDC model

System Specifications	Device Rating
SPV Panel	Maximum power = 0.6 kW Open circuit Voltage = 32 V Short circuit current = 19 A
Isolated converter switching frequency	24 kHz
DC link capacitor	372 μ F
Inverter	Three Phase MOSFET module, 600V, 10A
Field Programmable Gate Array (FPGA)	SPARTAN-3E XC3S100E controller
BLDC Motor	½ hp, 3000 rpm, 230V, 1.6A
Pump	Centrifugal pump

Table 1 describes the Specification of the proposed solar fed BLDC model; based on the Specification, the proposed prototype is developed and tested with the simulation results. Figure 5 represents the proposed Dynamic Energy Amendment Control (DEAC) strategy-based speed control of the BLDCM. Based on the different torque-speed of the BLDCM, performance is verified, and the results are given below.

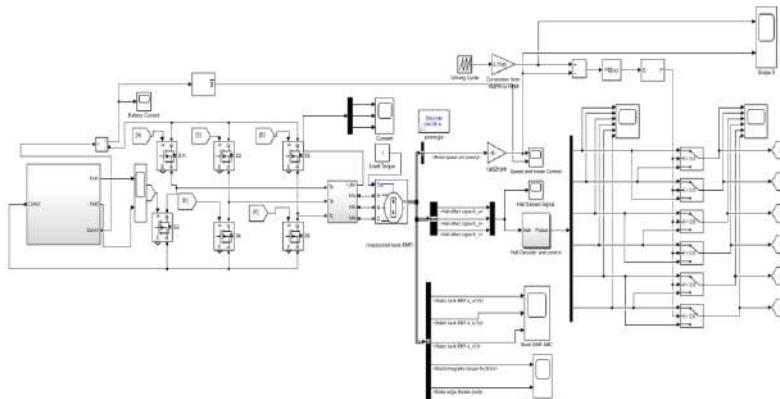


Fig 5. Final proposed system Simulink model

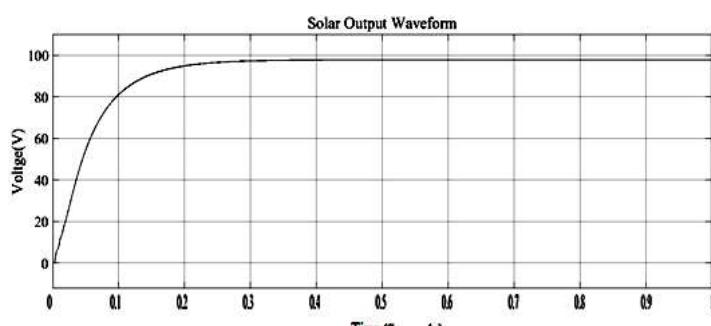


Fig 6. Solar Output Voltage

Figure 6 shows the solar power output voltage of the proposed model. In this system, an SPV generates the maximum voltage of 90 V. Figure 7 shows the DC-DC converter output voltage for the essential needs to run the BLDCM. The inverter needs the stabilized DC voltage, and the solar voltage will be improved by using the boost converter. The above waveform represents the solar power voltage which is increased up to 440 voltage. The proposed APO controller will stabilize the solar voltage at the time (0.15) seconds.

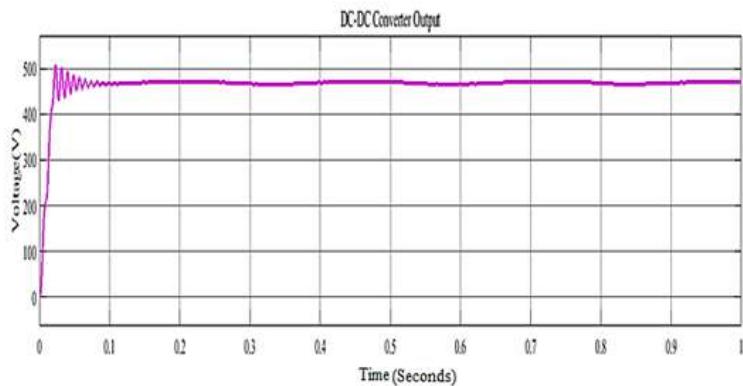


Fig 7. DC-DC converter waveform

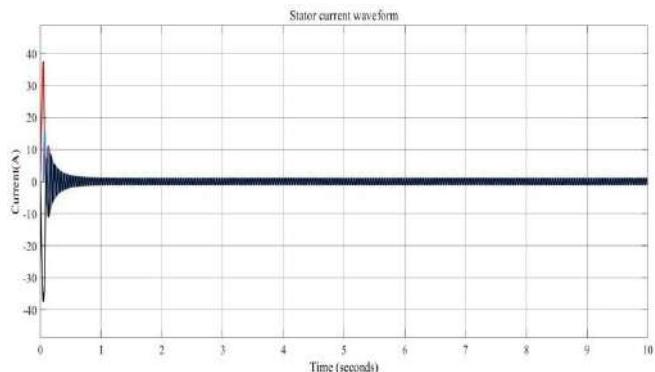


Fig 8. Output current of the proposed system

Load torque fluctuation is shown in Figure 8 by the current waveform for the BLDC motor; the initial current variation is large during the time period (0-2).

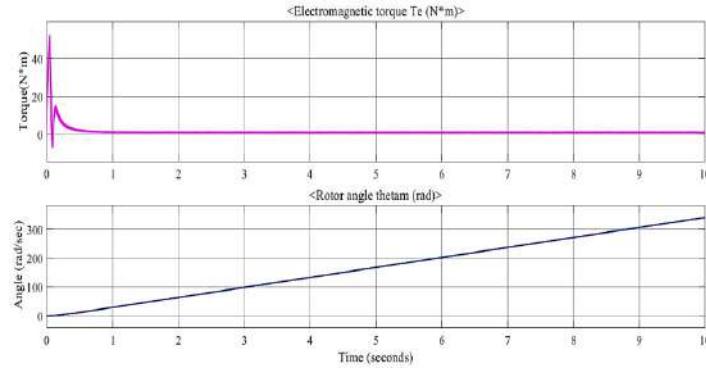


Fig 9. BLDC Motor Torque

Torque output from a brushless DC motor is seen in Figure 9. Applying the load torque for 0.2 seconds allows for a differential speed study of the suggested model. The torque value is used to adjust the speed and stator current.

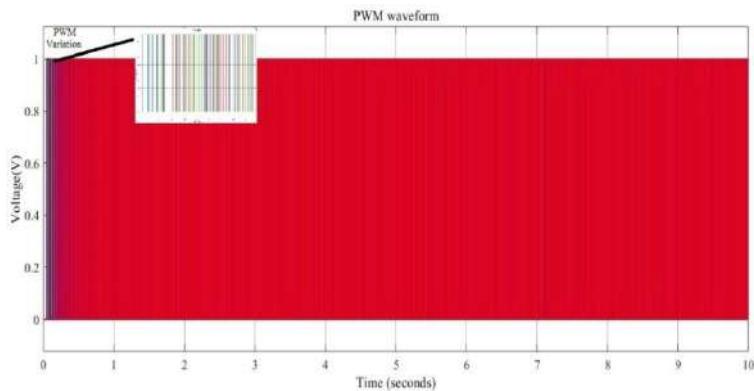


Fig 10. PWM waveform of the proposed controller

In order to keep the BLDC motor running at a constant speed, the suggested Dynamic Energy Amendment Control (DEAC) Controller would automatically adjust the PWM based on the load torque fluctuation. Figure 9 illustrates the varying torque under load, while Figure 10 shows PWM pulses superimposed on the load. As can be shown in Figure 11, the suggested Dynamic Energy Amendment Control (DEAC) Controller is able to analyse the speed stability of BLDC motors. Considering these factors, we determine that a driving cycle speed of 1500 RPM with a simultaneous torque of 1 N-m (0-0-1) is optimal.

The accompanying diagram illustrates the ideal speed that can be maintained with the help of the suggested DEAC controller by providing the proper feedback to the inverter. The performance of BLDCA is analysed by comparing many distinct metrics, such as the steady-state error, the recovery time, and the peak overshoot time, as shown in Figure 12. Simulation results are

organized in Table 2; it is clear that the DEAC-based controller outperforms the conventional Proportional Integral Derivative (PID) controller. Table 3 and Figure 17 show the characteristics of the control system under a variety of load torque settings. The DEAC controller that we suggest will be compared to the current system.

Table 2 The performance analysis of the BLDC motor

Parameters	PID	DEAC
ReferenceValue(RPM)	1500	1500
Peak Overshoottime (sec)	0.9564	0.5663
RecoveryOvershoot (%)	0.845	0.423
RecoveryTime(sec)	0.88	0.41
Steady StateError Value(rpm)	6.6	4.2
SteadyStateError (%)	0.66	0.36

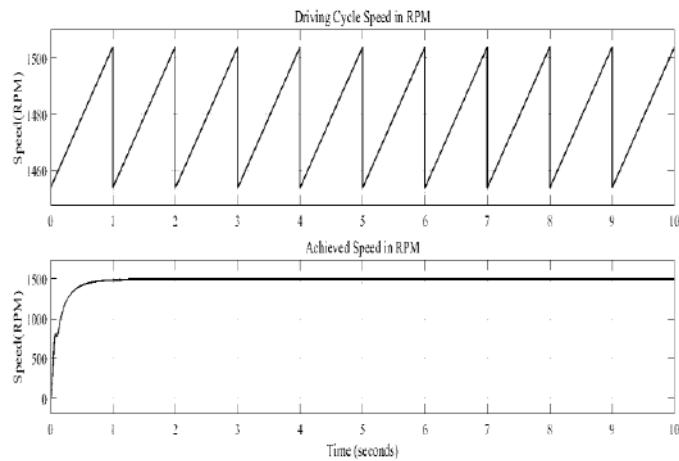


Figure 11. BLDC Motor speed

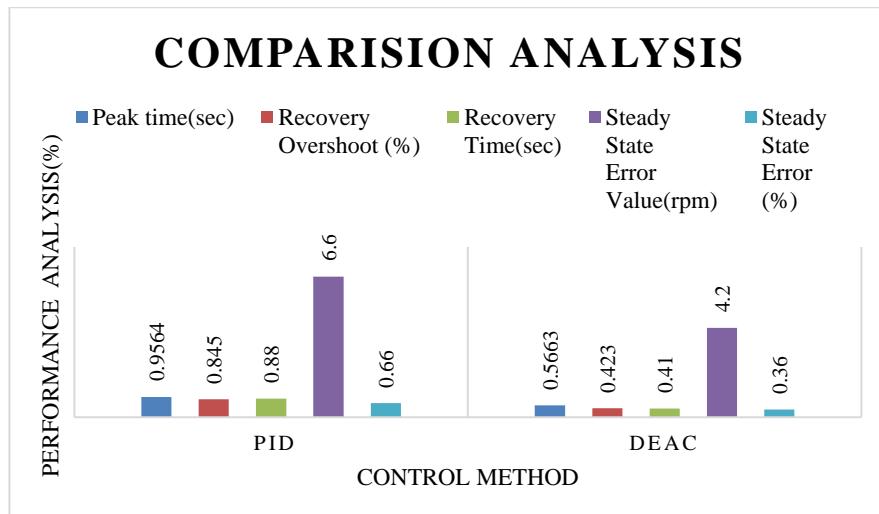


Fig 12. Comparison analysis

5 Conclusion

This paper presents a prototype model for controlling the speed of BLDCM through the Dynamic Energy Amendment Control (DEAC) method, and it is based on a controller based on the Field Programmable Gate Array (FPGA) (SPARTAN-3E XC3S100E). Using the Mat Lab 2017b software environment, the performance of each controller parameter has been checked and analysed in depth. The simulation results with this model show that the operating conditions for BLDC speed control are improved with the controller, leading to a steady-state value of 5.8% that is superior to that achieved with conventional controllers. The findings suggest that the proposed DEAC controllers' standout qualities include their ability to eliminate errors and drastically reduce steady-state error. Several load torque circumstances are used to validate the continuous and constant speed operations. The optimal steady-state value (RPM) for water pumping is achieved with the help of the cutting-edge Dynamic Energy Amendment Control (DEAC) control.

References

1. P. Mishra, A. Banerjee and M. Ghosh, "FPGA-Based Real-Time Implementation of Quadral-Duty Digital-PWM-Controlled Permanent Magnet BLDC Drive," in IEEE/ASME Transactions on Mechatronics, vol. 25, no. 3, pp. 1456-1467, June 2020, doi: 10.1109/TMECH.2020.2977859.
2. A. Sen and B. Singh, "Peak Current Detection Starting Based Position Sensorless Control of BLDC Motor Drive for PV Array Fed Irrigation Pump," in IEEE Transactions on Industry Applications, vol. 57, no. 3, pp. 2569-2577, May-June 2021, doi: 10.1109/TIA.2021.3066831.

3. L. Wang, Z. Q. Zhu, H. Bin and L. Gong, "A Commutation Error Compensation Strategy for High-Speed Brushless DC Drive Based on Adaline Filter," in IEEE Transactions on Industrial Electronics, vol. 68, no. 5, pp. 3728-3738, May 2021, doi: 10.1109/TIE.2020.2984445.
4. J. Cai and X. Zhao, "Synthetic Hybrid-Integral-Threshold Logic-Based Position Fault Diagnosis Scheme for SRM & BLDC Drives," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-8, 2021, Art no. 1500608, doi: 10.1109/TIM.2020.3031357.
5. H. Zhang and H. Li, "Fast Commutation Error Compensation Method of Sensorless Control for MSCMG BLDC Motor With Nonideal Back EMF," in IEEE Transactions on Power Electronics, vol. 36, no. 7, pp. 8044-8054, July 2021, doi: 10.1109/TPEL.2020.3030777.
6. L. Yang, Z. Q. Zhu, H. Bin, Z. Zhang and L. Gong, "Virtual Third Harmonic Back EMF-Based Sensorless Drive for High-Speed BLDC Motors Considering Machine Parameter Asymmetries," in IEEE Transactions on Industry Applications, vol. 57, no. 1, pp. 306-315, Jan.-Feb. 2021, doi: 10.1109/TIA.2020.3033821.
7. M. S. Trivedi and R. K. Keshri, "Evaluation of Predictive Current Control Techniques for PM BLDC Motor in Stationary Plane," in IEEE Access, vol. 8, pp. 46217-46228, 2020, doi: 10.1109/ACCESS.2020.2978695.
8. Arunkumar, U., Radhakrishnan, G., Boobalan, S., & Kishore, J. (2020). Power Factor Correction Using Sensorless Brushless DC Motor with Bridgeless Converter Topology. *Journal of Computational and Theoretical Nanoscience*, 17(4), 1796-1803.
9. P. M. de Almeida, R. L. Valle, P. G. Barbosa, V. F. Montagner, V. Ćuk and P. F. Ribeiro, "Robust Control of a Variable-Speed BLDC Motor Drive," in IEEE Journal of Emerging and Selected Topics in Industrial Electronics, vol. 2, no. 1, pp. 32-41, Jan. 2021, doi: 10.1109/JESTIE.2020.3035055.
10. P. Srivastava and V. K. Tiwari, "Speed Control of BLDC Motor fed on Solar PV Array using Particle Swarm Optimization," 2020 9th International Conference System Modeling and Advancement in Research Trends (SMART), 2020, pp. 392-397, doi: 10.1109/SMART50582.2020.9337111.
11. B. Tian, Q. An and M. Molinas, "High-Frequency Injection-Based Sensorless Control for a General Five-Phase BLDC Motor Incorporating System Delay and Phase Resistance," in IEEE Access, vol. 7, pp. 162862-162873, 2019, doi: 10.1109/ACCESS.2019.2950256.
12. Pavana and U. Vinatha, "FPGA based experimental evaluation of BLDC motor drive fed from coupled inductor based bridgeless SEPIC," 2020 IEEE International Conference on Power Electronics, Smart Grid and Renewable Energy (PESGRE2020), 2020, pp. 1-6, doi: 10.1109/PESGRE45664.2020.9070453.
13. Alqahtani, Abdulrahman Saad, et al. "Solar PV fed brushless drive with optical encoder for agriculture applications using IoT and FPGA." *Optical and Quantum Electronics* 54.11 (2022): 1-14.

14. F. Pereira and L. Gomes, "The IOPT-Flow Modeling Framework Applied to Power Electronics Controllers," in IEEE Transactions on Industrial Electronics, vol. 64, no. 3, pp. 2363-2372, March 2017, doi: 10.1109/TIE.2016.2620101.
15. W. Lee, J. H. Kim, W. Choi and B. Sarlioglu, "Torque Ripple Minimization Control Technique of High-Speed Single-Phase Brushless DC Motor for Electric Turbocharger," in IEEE Transactions on Vehicular Technology, vol. 67, no. 11, pp. 10357-10365, Nov. 2018, doi: 10.1109/TVT.2018.2866779.
16. R. K. Megalingam, S. R. R. Vadivel, B. T. Pula, S. Reddy Sathi and U. S. C. Gupta, "Motor Control Design for Position Measurement and Speed Control," 2019 International Conference on Communication and Signal Processing (ICCSP), 2019, pp. 0405-0409, doi: 10.1109/ICCSP.2019.8698016.
17. S. Chen, G. Liu and L. Zhu, "Sensorless Control Strategy of a 315 kW High-Speed BLDC Motor Based on a Speed-Independent Flux Linkage Function," in IEEE Transactions on Industrial Electronics, vol. 64, no. 11, pp. 8607-8617, Nov. 2017, doi: 10.1109/TIE.2017.2698373.
18. A. Darba, F. De Belie, P. D'haese and J. A. Melkebeek, "Improved Dynamic Behavior in BLDC Drives Using Model Predictive Speed and Current Control," in IEEE Transactions on Industrial Electronics, vol. 63, no. 2, pp. 728-740, Feb. 2016, doi: 10.1109/TIE.2015.2477262.
19. W. Li, J. Fang, H. Li and J. Tang, "Position Sensorless Control Without Phase Shifter for High-Speed BLDC Motors With Low Inductance and Nonideal Back EMF," in IEEE Transactions on Power Electronics, vol. 31, no. 2, pp. 1354-1366, Feb. 2016, doi: 10.1109/TPEL.2015.2413593.
20. D. Das, N. Kumaresan, V. Nayanar, K. Navin Sam and N. Ammasai Gounden, "Development of BLDC Motor-Based Elevator System Suitable for DC Microgrid," in IEEE/ASME Transactions on Mechatronics, vol. 21, no. 3, pp. 1552-1560, June 2016, doi: 10.1109/TMECH.2015.2506818.
21. M. Baszynski and S. Pirog, "A Novel Speed Measurement Method for a High-Speed BLDC Motor Based on the Signals From the Rotor Position Sensor," in IEEE Transactions on Industrial Informatics, vol. 10, no. 1, pp. 84-91, Feb. 2014, doi: 10.1109/TII.2013.2243740.
22. R. Horvat, K. Jezernik and M. Čurkovič, "An Event-Driven Approach to the Current Control of a BLDC Motor Using FPGA," in IEEE Transactions on Industrial Electronics, vol. 61, no. 7, pp. 3719-3726, July 2014, doi: 10.1109/TIE.2013.2276776.
23. N. Milivojevic, M. Krishnamurthy, Y. Gurkaynak, A. Sathyam, Y. Lee and A. Emadi, "Stability Analysis of FPGA-Based Control of Brushless DC Motors and Generators Using Digital PWM Technique," in IEEE Transactions on Industrial Electronics, vol. 59, no. 1, pp. 343-351, Jan. 2012, doi: 10.1109/TIE.2011.2146220.
24. Kumar, U. A., & Ravichandran, C. S. (2022). Upgrading the quality of power using TVSS device and PFC converter fed SBLDC motor. *Arabian Journal for Science and Engineering*, 47(8), 9345-9359.
25. M. Baszynski and S. Pirog, "A Novel Speed Measurement Method for a High-Speed BLDC Motor Based on the Signals From the Rotor Position Sensor," in

IEEE Transactions on Industrial Informatics, vol. 10, no. 1, pp. 84-91, Feb. 2014,
doi: 10.1109/TII.2013.2243740.

Arabic Sentiment Analysis of YouTube Comments using Deep Learning Model

*Mohammed Alkoli¹, B. Sharada²

¹Research scholar

Department of Studies in Computer Science

University of Mysore

Manasa Gangothiri, Mysuru, Karnataka-570006,

India,

²Professor

Department of Studies in Computer Science

University of Mysore,

Manasa Gangothiri, Mysuru, Karnataka-570006,

India,

²sharada34s@outlook.com

* Corresponding Author: alkolim528@gmail.com

Abstract. With the present prevalence of social media websites, acquiring user priorities has become an essential chore for assessing their online habits and behaviors. Because Arabic is amongst the most prevalent languages throughout the globe as well as the quickest expanding language on the Internet, we are motivated to create dependable automated technologies. We developed a sentiment analysis on YouTube comments premised on manually collected Arabic comments in this research. This work was divided into four stages: preprocessing, feature extraction, optimum feature selection, as well as classification. Tokenization, stemming, and stop word removal are all part of the preprocessing process. We designed two unique features, Proposed Term Frequency Inverse Document Frequency (Proposed TD-IDF) and improved word2vect-based features, to improve feature extraction with the conventional Bag of words feature extraction. In addition, for optimal feature selection, we introduced a new optimization technique called the Self-Improved Honey Badger Algorithm (SIHBA).

Keywords: Term Frequency Inverse Document Frequency (TD-IDF), Improved Honey Badger Algorithm (SIHBA), Bidirectional –LSTM.

1. Introduction

The volume of Arabic material produced for websites as well as social media platforms has increased dramatically during the past couple of years [1]. Currently, people interact online and express their thoughts through comments on online networking platforms like Facebook, Instagram, as well as Google+. In terms of video distribution, YouTube is

generally regarded as the best. It's the world's biggest platform where users may upload and make suggestions on videos to convey their opinions [2]. Sentiment analysis seems to be vital in social networking surveillance due to the impact of media blogson democratic governance, advancement, diplomacy, as well as enterprise. It does this by allowing an insight into the general public's view on subjects that show up in a diverse range of posts, from blogs about politicians to user reviews. Recognizing the motivation and perspective behind a comment on any topic makes it simple to plan & strategize, which improves services [3]. Advanced quantitative, as well as machine learning frameworks, have been used in sentiment analysis (SA), a crucial location of natural language processing (NLP), to evaluate feelings, moods, as well as thoughts in a diverse array of contexts, along with consumer fulfillment, market reach, price trends, and public support for democratic choices and events. Systems for sentiment analysis have lately received increased attention as social media as well as Web applications havegrown in popularity. According to the rising number of Netizens who use Arabic, which makes up approximately 5% of all users globally [4,5,6], the emphasis is on feelings conveyed in that language. According to SA, it is generally vital to recognize the four components that make up an entity: its perspective, the viewpoint holders, and their feeling [7]. Finding the sentiment (positivity, negativity, or neutrality) of an author is the goal of Arabic sentiment analysis (ASA), also known as opinion extraction. Tweets on various social media sites including Fb page, YouTube, Insta, as well as Twitter reflect a wide range of perspectives. Recent years have seen a rise in interest in this area of study, particularly in English. Only a few works have utilized ASA thus far, despite the Arabic language being considered the most helpful language on social networking sites [8,9]. A lot of studieshavebeen performed on Arabic sentiment analysis during the past few years (ASA). Technologies for sentiment analysis were conducted in many distinct fields utilizing a variety of strategies and machine learning techniques [10]. Regarding the diversity of preceding attempts, these are generally limited by the corpora size, and restrict the generalization of the machine learning model, especially while deep neural networks are being used [11,12]. Emotions in daily encounters with people are self-explanatory and effective [21].

Despite these we have developed anArabic sentiment analysis of YouTube comments, which uses Bi-LSTM to classify the comments based on the selected optimal features and the contributions of our proposed work, as given below:

- ❖ Arabic sentiment analysis was conducted on manually collected YouTube comments which use Bi-LSTM to provide better classification results.
- ❖ With the conventional Bag of Words feature extraction, two novel features such as proposed TD-IDF and improved word2vec-based features were developed to get higher classification results.
- ❖ A Self Improved Honey Badger Algorithm was developed in this work, which reduces the computational time by selecting the optimal features from those extracted features.

The following is how this paper was structured: Section 2 contains several papers relevant to Arabic sentiment analysis (ASA). Section 3, has a brief overview of our proposed methodology, Section 4 comprises the findings of our work, Section 5 contains the conclusion of our work, and the following section contains the references to this work.

2. Literature survey

Some of the works which are related to the ASA were reviewed here.

Hasna et al [13] proposed an Arabic Sentiment Analysis approach that integrates an Arabic BERT tokenizer instead of using a basic BERT Tokenizer. For hyper-parameter optimization, this work uses the random search method. Abdullah et al [14] suggested a machine learning(ML) strategy for analyzing Twitter Arabic postings. Word2Vec was utilized in this approach for the embedding of the word that was the significant supply of characteristics.

A unique deep learning framework for Arabic language sentiment analysis was offered by Abubakr et al. [15] and is built on two layers of LSTM to preserve long-term connectivity and one layer of CNN architecture for extracting features. The SVM classifier delivers the end categorization utilizing the feature maps learned by CNN as well as LSTM. To optimize Arabic Sentiment Analysis, Hanane et al. [16] created an effective Bidirectional LSTM Network (BiLSTM) by employing Forward-Backward encapsulation of contextualized data from Arabic characteristic sets. A unique deep learning-dependent approach that concurrently computes contextualized embeddings at the character, phrase, as well as sentence levels, as presented by MOHAMMED et al. [17], employs a basic positional binary embedding scheme (PBES).

Since Arabic is the fourth most widely spoken language and the language with the fastest expanding user base on the Internet, where usage is increasing at a pace of 6.6% annually, Arabic sentiment analysis is regarded as a significant research endeavour in the field of sentiment analysis. However, it is thought to be a difficult challenge to perform sentiment analysis for Arabic as a Morphological Rich Language (MRL). Each Arabic root term has numerous derivatives, each of which has virtually entirely distinct connotations and polarities depending on the locality and dialect. Derivatives can occasionally have various interpretations and polarity.

3. Methodology

With the current level of ubiquity of social media websites, obtaining the user's preferences automatically became a crucial task to assess their tendencies and behaviors online. Arabic language as one of the most spoken languages in the world and the fastest growing language on the Internet motivates us to provide reliable automated tools that can perform sentiment analysis to reveal users' opinions. In this research work, a novel Arabic sentiment analysis technique on YouTube Comments will be developed. The projected model encapsulates four major phases:

In the preprocessing step, stop words, stemming, and tokenization will be used to preprocess the acquired raw data. The second phase involved extracting features from the pre-processed data, including Bag of Words, Proposed Term Frequency Inverse Document Frequency (TD-IDF), and improved word2vect-based features. In the third phase, the Self-improved Honey Badger Algorithm (SIHBA) will be used to choose the features that are

most ideal from those that have already been chosen. Finally, the Bidirectional Long Short-Term Memory will be used for the review classification (Bi-LSTM). The SIHBA model's optimally chosen features will be used to train the Bi-LSTM classifier. Bidirectional long-term dependencies between time steps of time series or sequence data are learned via a bidirectional LSTM (BiLSTM) layer. When you want the network to learn from the entire time series at each time step, these dependencies can be helpful. The final results will exhibit whether the collected review points to a positive, neutral or negative sentiment.

Fig.1 manifests the architecture of our proposed work.

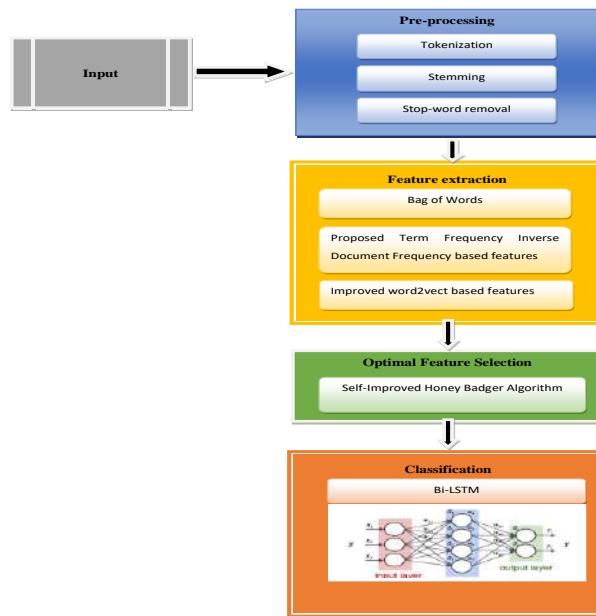


Fig.1.The architecture of the Proposed Arabic Sentiment analysis Method

3.1 Pre-Processing

Preprocessing is indeed a vital step in computer vision. To render the input information convenient to read and utilize, we used three pre-processing strategies in our research: tokenization, stemming, as well as stop word removal. Each process would be described in full below.

3.1.1 Tokenization

This stage partitions the files into words/terms and then turns those into a bag of words. This would be accomplished through the use of unigrams, and bigrams, including trigrams, which are described as patterns of just one, two, or three contiguous words drawn from a list of tokens. We primarily employed unigrams, except in a few cases where two-word phrases were required.

3.1.2 Stemming

Stemming has been used to lessen the count of words (dimension reduction), computation difficulty, and overall operating speed. Stemming technologies function by deleting words' suffixes as well as prefixes.

3.1.3 Stop-word removal

Stop words are first compiled into a list, after which they are eliminated. Even though they are used more often, these terms typically have no clear meaning. Stop-words should be eliminated so that we may focus on the words that have the most significant meanings.

3.2 Feature Extraction

This feature extraction approach minimizes the count of characteristics in a dataset by generating new ones from existing ones, which enhances both training speed as well as accuracy. Bag of Words, Proposed Term Frequency Inverse Document Frequency (TD-IDF) based, as well as improved word2vect based features have been extracted to decrease overfitting & training time.

3.2.1 Bag of Words (BoW)

A bag of words seems to be a text-structuring approach in Natural Language Processing. In technical words, it is a feature retrieval approach using text data. This method is indeed a basic yet adaptable method of retrieving characteristics from documents. The bag - of - words is a textual portrayal of the incidence of words inside a document. We only count words and ignore grammatical subtleties and word order. The term "bag of words" refers to the fact that any data on the sequence or structure of words in the document is deleted. The model simply cares about whether identifiable terms appear in the document, not where they appear.

One of the most significant issues with text is that it's very chaotic as well as unorganized, while machine learning schemes favor organized, well-defined fixed-length input data. Fortunately, we can turn variable-length texts into fixed-length vectors by using the Bag-of-Words approach. Furthermore, the ml algorithms deal with mathematical information instead of textual information at a much finer level. To be more exact, we turn a sentence into its vector of integers utilizing the bag-of-words (BoW) mechanism.

3.2.2 Proposed Term Frequency Inverse Document Frequency-based features

In the present vector space design, the TF-IDF methodology is by far the most often utilized feature word weight computation tool. It is split into 2 parts: the frequency of words as well as the frequency of inverted texts. The number of repetitions of a specific term in the document is referred to as its word frequency. The reverse file frequency indicates the measurement of a word's overall relevance. The frequency of a word's inverse gets split by the overall count of files, which would be divided by the count of files having the phrase, as well as the quotient logarithmic is calculated. The word frequency (TF), as well as inverse text frequency (IDF) equations, would be as follows:

$$tf_{a,b} = \frac{N_{a,b}}{\sum_c N_{c,b}} \quad (1)$$

$N_{a,b}$ indicates the count of instances of the term T_a in the file d_b , while $\sum_c N_{c,b}$ representing the total of all instances in the file d_b .

$$idf_a = \log \frac{|D_{count}|}{|\{b : T_a \in d_b\}|} \quad (2)$$

$|D_{count}|$ denotes the total count of documents in the corpora, and $\{b : T_a \in d_b\}$ signifies the count of files that include the term t_i . Whereas if the term wasn't in the corpora, the dividend will be 0. Generally, employ $1 + |\{b : T_a \in d_b\}|$.

$$tfidf_{a,b} = \frac{tf_{a,b} \times idf_a}{\sqrt{\sum_{T_a \in d_b} [tf_{a,b} \times idf_a]^2}} \quad (3)$$

$tfidf_{a,b}$ represents the weight of the term T_a . It is shown that a large word frequency in a specific document, as well as a lower word frequency in the overall file group, can result in a higher-weight TF-IDF.

We have proposed the following formula to overcome the limitations of the conventional TF-IDF, and our proposed TF-IDF was given in (4)

$$TFIDF = \log \frac{N_{doc}}{N_b} \times \frac{tf \times idf}{\sum_{T \in d} [tf \times idf]^2} \quad (4)$$

N_{doc} indicates the overall count of documents.

N_b seems to be the count of documents containing the feature phrase.

d seems to be the total count of documents in the corpora.

T represents the count of documents that include the term.

TF = Word Frequency of Word, IDF = Inverse Text Frequency

3.2.3 Improved word2vect-based features

Word2vec seems to be a linguistic model founded on neural networks. There are 2 training topologies available: continuous bag-of-words (CBOW) as well as skip-gram (SG). CBOW's training speed was quicker than skip-gram's, hence we employed an improved CBOW in our work. CBOW anticipates target words depending on the context of the text. Contextual text is indeed a word series with a certain window size:

$$CT(\omega_j) = \{\omega_{j-k}, \omega_{j-k+1}, \dots, \omega_{j-1}, \omega_{j+1}, \omega_{j+2}, \dots, \omega_{j+k}\} \quad (5)$$

The following steps demonstrate how our improved word2vec functions.

1. Determine the window size and vector space dimension. The window size is 5 and the vector dimension is calculated by using word 2 vector.
2. Scanning all of the training corpus articles to construct the vocabulary
3. Set up the word vector as well as a neural network.
4. Retrieve the training sequences from the article sentences.

5. Train the neural net with the Improved CBOW Algorithm and then use cosine similarity to compare the similarity of two sentences.

$$CT(\omega_j) = \frac{\{\omega_{j-k}, \omega_{j-k+1}, \dots, \omega_{j-1}, \omega_{j+1}, \omega_{j+2}, \dots, \omega_{j+k}\}}{\cos(y, z)} \quad (6)$$

Cosine similarity is indeed a statistic that may be used to determine how identical data items are regardless of size. Data items in a dataset were regarded as vectors in cosine similarity. Another equation for calculating the cosine similarity of 2 vectors was

$$\text{Cos}(y, z) = y \cdot z / \|y\| * \|z\| \quad (7)$$

where,

$y \cdot z$ denotes the dot product of 'y' as well as 'z' vectors

$\|y\|, \|z\|$ signifies the sum of the lengths of the two vectors 'y' as well as 'z'

$\|y\| * \|z\|$ denotes the product of two vectors 'y' as well as 'z'

k represents the window size

The target word gets denoted by ω_j .

ω_i signifies the weight function calculated via the sine map. For the sine map

$$k_{l+1} = \sin(\pi k_l)$$

3.3 Optimal feature selection

The effectiveness of feature selection has a substantial impact on classification outcomes, hence it is a significant concern in sentiment classification. The Self-Improved Honey Badger Algorithm was used in the work for optimum feature selection.

In theory, HBA is indeed a global optimization method since it includes both global search stages. Method 1 presents a pseudo-code for the proposed procedure, which includes population initialization, population assessment, then parameter updates. The steps of the Self Improved HBA (SIHBA) are outlined mathematically as follows.

Step 1: Set the parameters maximum iteration (M_{\max}), population size (N), constant (c), β .

Step 2: Initialize population with random positions.

Step 3: Compute the fitness of each honey badger position.

Step 4: Save the best position p_{prey} and assign fitness to f_{prey} .

Step 5: While $M \leq M_{\max}$ do

Step 6: As per proposed methodology update the density factor using the Eq. (8). Where D_{circle} = Diameter of the circles' used area by the honey badger.

$$\delta = c \times \exp \times D_{circle} \left(\frac{-M}{M_{\max}} \right) \quad (8)$$

Step 7: For i=1 to N do

Step 8: Compute the intensity by using the Eq. (9). Here, R_2 denotes a random number between 0 & 1 and d_q represents the distance between both the prey as well as the q^{th} badger.

$$I_q = R_2 \times \frac{s}{4\pi d_q^2}, \quad (9)$$

Step 9: if $e < 0.5$, then (e is a random number among 0 and 1)

Step 10: Update the position using Eq. (10).

$$p_{new} = p_{prey} + f \times \beta \times I \times p_{prey} + f \times R_3 \times \delta \times d_q \times [\cos(2\pi R_4) \times [1 - \cos(2\pi R_5)]] \quad (10)$$

Where, p_{prey} represents the prey's perfect location identified thus far - in other words, the global perfect location $\beta \geq 1$ (default = 6) represents the honey badger's capacity to obtain food. The term, $R_3, R_4, and R_5$ are the three separate random integers ranging from 0 to 1, and f serves as a flag that changes the search position. It is decided by Eq (11).

$$f = \begin{cases} 1 & \text{if } R_6 \leq 0.5 \\ -1 & \text{else} \end{cases}, \quad R_6 \text{ is a random number between 0 and 1.} \quad (11)$$

A honey badger depends substantially on scent strength I of prey p_{prey} , proximity between the badger as well as prey d_q , and time-varying exploration impact factor α during digging. Furthermore, while digging, a badger may encounter any disruption F, allowing it to locate even greater prey.

Step 11: else

Step 12: As per proposed contribution, update the position using Eq. (12) and Eq. (13).

$$p_{new} = \frac{p_{prey} + f \times R_7 \times \delta \times d_q \times \eta_\sigma}{m} \quad (12)$$

$$\text{Median} = l_m + \frac{\frac{N_f}{2} - C_f}{m_f} \times \sigma \quad (13)$$

Here, l_m denotes the lower limit of the median class, N_f denotes the sum of all frequencies, m_f indicates the median class frequency, σ symbolizes the median class width, C_f signifies the cumulative frequency of the class pending the median class, η_σ =Weight integer randomly selected between 0 to 2, R_7 =Random number between 0 and 1 and this randomness is estimated by using a circle map

$c_{K+1} = c_K + 0.5 - \frac{1.1}{\pi} \sin(2\pi c_K)$, and δ denotes the form of time-varying search influence.

Step 12: end if

Step 13: Compute new positions and assign to f_{new} .

Step 14: Set $p_i = p_{new}$ and $f_i = f_{new}$.

Step 15: end if

Step 16: if $f_{new} \leq f_{prey}$ then

Step 17: Set $p_{prey} = p_{new}$ and $f_{prey} = f_{new}$.

Step 18: end if

Step 19: end for

Step 20: end while stopping criteria satisfied.

Step 21: Return p_{prey} .

3.4 Sentimental Classification

The final stage of our research is classification, which categorizes comments as positive, negative, or neutral depending on the specified features. In our work, the Bi-LSTM classifier will be trained using the SIHBA model's optimally selected features. By incorporating memory Z as well as the gate architecture, the LSTM network overcomes the long-term dependence issue which occurs in traditional RNNs. An input gate determines which information may be sent to the cell and is described as follows:

$$i_x = \alpha(W_i \cdot [h_{x-1}, x_x] + b_i) \quad (17)$$

The forget gate determines whether input data should be ignored from prior memory and also is specified as:

$$f_x = \alpha(W_f \cdot [h_{x-1}, x_x] + b_f) \quad (18)$$

Depending on equations (19) as well as (20), the control gate regulates the updating of cell conditions from Z_{x-1} to Z_x .

$$\overset{\square}{Z}_x = \tanh(W_z \cdot [h_{x-1}, x_z] + b_z) \quad (19)$$

$$Z_x = f_x * Z_{x-1} + i_x * \overset{\square}{Z}_x \quad (20)$$

The output gate seems to be in charge of producing output while updating the concealed vector $ht-1$. This procedure is as follows:

$$o_x = \alpha(W_o \cdot [h_{x-1}, x_x] + b_o) \quad (21)$$

$$h_x = o_x * \tanh(C_x)$$

In equations (17) to (21), α is indeed the sigmoid activating function, Ws seem to be the appropriate weight matrices, while tanh is being utilized to adjust the values between -1 and 1.

We looked at a single LSTM layer, one bidirectional LSTM layer, plus 3 completely linked layers with 8, 64, as well as 8 units, correspondingly. The Bi-LSTM output layer determines if the comment is positive, negative, or neutral.

4 Results and Discussion

This proposed work is implemented in Python 3.7 and pycharm 2021.3. The manually collected dataset was used in our work and the number of samples considered in this work are 433. The performance of our proposed SIHBA was compared with the conventional algorithms such as HBA, DHOA, DO, SSOA, and DOX, and the results were given below. Table 1 shows the Bi-LSTM classifier hyperparameter.

Table 1. Bi-LSTM Hyperparameter

Hyperparameter	Value
batch_size	10
verbose	1
epochs	6

The positive performance matrices such as accuracy, precision, sensitivity, and specificity of our Self Improved Honey Badger optimization were estimated for (60-90) Learning Percentages and are compared with existing techniques such as HBA, DHOA, DO, SSOA, and DOX which is shown in fig 2.

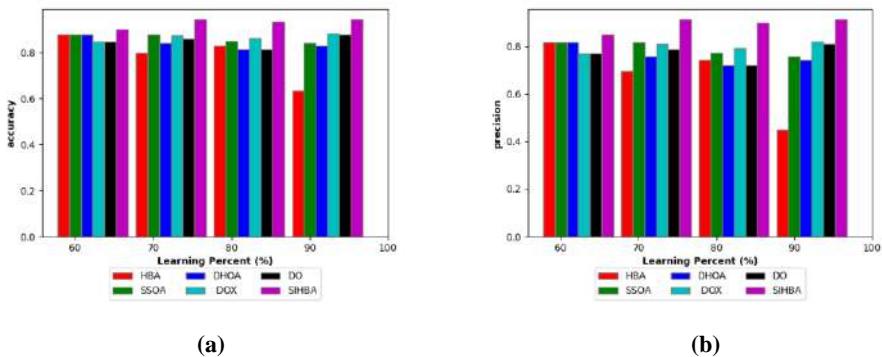


Fig.2. Comparison of Performance measures such as (a) Accuracy (b) Precision

When HBA achieves the accuracy rate of 0.88, 0.79, 0.8, and 0.6 for 60-90 LPs while our Self Improved Honey Badger Algorithm (SIHBA) achieves the rate of 0.9, 0.96, 0.95, 0.97. When conventional HBA achieves the precision rate of 0.81, 0.7, 0.76, 0.5, our SIHBA achieves a rate of 0.88, 0.94, 0.9, 0.92, for 60-90 LPs, which is higher than other techniques.

When HBA achieves the accuracy rate of 0.88, 0.79, 0.8, and 0.6 for 60-90 LPs while our Self Improved Honey Badger Algorithm (SIHBA) achieves the rate of 0.9, 0.96, 0.95,

0.97. When conventional HBA achieves the precision rate of 0.81, 0.7, 0.76, 0.5, our SIHBA achieves a rate of 0.88, 0.94, 0.9, 0.92, for 60-90 LPs, which is higher than other techniques.

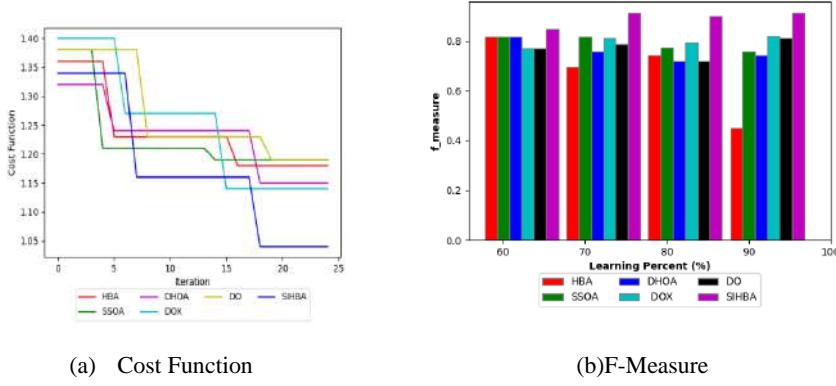


Fig.3. Comparison of the measures such as (a) Cost Function and (b) F-Measure

The cost function for 0-25 iterations was also calculated for our proposed SIHBA and the results were compared with conventional techniques, which are shown in fig 3(a). Although the cost function of our proposed SIHBA was 1.34 for 0-5 iterations, which is higher than DHOA, for 5-15 iterations its value gets reduced and reaches the rate of 1.05 for iterations 20-25, which proves the effectiveness of our proposed SIHBA. With that F-measure values for 60-100LPs were also calculated for our proposed SIHBA, which is compared with conventional techniques, which is shown in fig 3(b). For 90 LP, HBA and DHOA achieve a lower rate of 0.43, and 0.74, while our SIHBA achieves a higher value of 0.94.

The analysis such as best, worst, mean, median, and standard deviation (std) was calculated for our proposed SIHBA and the results were compared with the conventional algorithms such as HBA, DHOA, DO, and SSOA, DOX which is shown in table 1. When conventional HBA, achieves the best and worst values of 0.122, 0.366 while our proposed SI-BHA method achieves the lower values of 0.05, 0.09. Similarly, the mean and std values of our proposed SIHBA were low compared to other methods, which shows that our method can provide higher performance. To evaluate the effectiveness of our proposed SIHBA, it is worked in different scenarios, which is shown in Table 2. While using the conventional TD-IDF method, the accuracy level were only 0.804 while our proposed SIHBA achieves values of 0.94 and 0.913.

Table 2. Performance matrix comparison of the proposed method with different scenarios.

Performance matrices	Proposed Method with Conventional TD-IDF	Prop Method with ConvWord2Vec	Prop without Optimization	Proposed SIHBA
accuracy	0.804598	0.850575	0.835249	0.942529
precision	0.706897	0.775862	0.752874	0.913793
f_measure	0.706897	0.775862	0.752874	0.913793

The performances of the classifiers such as SVM, Naïve Bayes, Decision Tree, RNN, and LSTM were calculated and the results were compared with our proposed SIHBA, which is shown in Table 3. When SVM, Naïve Bayes classifiers achieve the accuracy and precision rates of 0.725, 0.587, and 0.623, 0.435, our proposed SIHBA achieves the rate of 0.942 and 0.913

Table 3. Performance matrices comparison of different classifiers with proposed SIHBA

Performance matrices	SVM	Naïve Bayes	Decision Tree	RNN	LSTM	SIHBA
Accuracy	0.725264	0.623439	0.623439	0.717579	0.846302	0.942529
Precision	0.587896	0.435159	0.435159	0.576369	0.769452	0.913793
F-measure	0.587896	0.435159	0.435159	0.576369	0.769452	0.913793

5. Conclusion

The sentiment analysis in texts has mostly concentrated on the English language. Because of the intricacy of the Arabic language as well as its grammatical elements that differ from those present in English, existing research has been unable to be applied to Arabic situations, restricting progress in Arabic sentiment analysis. In this paper, we created an Arabic Sentiment Analysis based on manually gathered YouTube comments. Our study was divided into three stages: preprocessing, feature extraction, optimum feature selection, and classification. Tokenization, stemming, and stop word removal are all part of this preprocessing stage. To reduce the training and computational time, optimal features were selected from these extracted features by proposing a Self Improved Honey Badger Algorithm (SIHBA). The limitation of the proposed work is to concentrate on the large dataset. In the future work, larger dataset can be considered.

Reference

- [1] AlhumoudS.O., Al Wazrah, A. A.: Arabic sentiment analysis using recurrent neural networks: a review. Artificial Intelligence Review. 1-42 (2021).
- [2] Wadhwani,S., Richhariya,P.,Soni,A.: Analysis & Implementation of Sentiment Analysis of User YouTube Comments.(2022).
- [3] Babu,N.V., Kanaga,E.: Sentiment analysis in social media data for depression detection using artificial intelligence: A review. SN Computer Science. 3, 1-20(2022).
- [4] Alharbi, A., Taileb,M.,Kalkatawi, M.: Deep learning in Arabic sentiment analysis: An overview. Journal of Information Science. 47, 129-140 (2021).
- [5] Li, W., Shao,W., JiS., CambriaE.: BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis. Neurocomputing. 467,73-82(2022).
- [6] Farha,I. A.,Magdy, W. A.: comparative study of effective approaches for Arabic sentiment analysis. Information Processing & Management.58, 102438 (2021).
- [7] Ghallab,A.,Mohsen A., and Ali, Y.: Arabic sentiment analysis:A systematic literature review. Applied Computational Intelligence and Soft Computing. (2020).
- [8] Zahidi,Y., El Younoussi,Y., Al-Amrani, Y.: A powerful comparison of deep learning frameworks for Arabic sentiment analysis. International Journal of Electrical & Computer Engineering. 11(2021).

- [9] Oussous, A., Benjelloun,F.Z.,LahcenA.A., Belfkih, S.: ASA: A framework for Arabic sentiment analysis. *Journal of Information Science*. 46, 544-559 (2020).
- [10] Alsayat,A., Elmitwally, N.: A comprehensive study for Arabic sentiment analysis (challenges and applications). *Egyptian Informatics Journal*. 21, 7-12 (2020).
- [11] Alroobaea,R.: Sentiment Analysis on Amazon Product Reviews using theRecurrent Neural Network (RNN). *International Journal of Advanced Computer Science and Applications*. 13(2022).
- [12] Mohammed,A., Kora, R.: Deep learning approaches for Arabic sentiment analysis, *Social Network Analysis and Mining*. 9, 1-12 (2019).
- [13] Chouikhi,H.,Chniter,H., Jarray, F.: Arabic sentiment analysis using BERT model. In *International Conference on Computational Collective Intelligence*. 621-632 (2021).
- [14] Al-Hashedi, A., Al-Fuhaidi, B., Mohsen, A.M., Ali, Y., Gamal Al-Kaf, H.A., Al-Sorori,W.,Maqtary,N.: Ensemble classifiers for Arabic sentiment analysis of social network (Twitter data) towards COVID-19-related conspiracy theories. *Applied Computational Intelligence and Soft Computing*. (2022).
- [15] Ombabi, A.H., OuardaW., Alimi,A.M.: Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks. *Social Network Analysis and Mining*. 10, 1-13(2020).
- [16] Elfaik, H.: Deep bidirectional lstm network learning-based sentiment analysis for arabic text, *Journal of Intelligent Systems*. 30, 395-412(2021).
- [17] El-Affendi, M.A., Alrajhi,K., Hussain,A.: A novel deep learning-based multilevel parallel attention neural (MPAN) model for multidomain arabic sentiment analysis. *IEEE Access*, 9, 7508-7518, (2021).

Learning Movement Patterns for Improving the Skills of Beginner Level Players in Competitive MOBAs *

Dane Brown¹[0000-0001-7395-7370] and Jonah Bischof¹

Rhodes University, Drosty Rd, Grahamstown, 6140, South Africa

d.brown@ru.ac.za

[https:](https://www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/)

//www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/

Abstract. League of Legends is a Massively Multiplayer Online Battle Arena (MOBA) — a form of online competitive game in which teams of five players battle to demolish the opponent’s base. Expert players are aware of when to target, how to maximise their gold, and how to make choices. These are some of the talents that distinguish them from novices. The Riot API enables the retrieval of current League of Legends game data. This data is used to construct machine learning models that can benefit amateur players. Kills and goals can assist seasoned players understand how to take advantage of micro and macro teams. By understanding how professional players differ from novices, we may build tools to assist novices’ decision-making. 19 of 20 games for training a Random Forest (RF) and Decision Tree (DT) regressor produced encouraging results. An unseen game was utilised to evaluate the impartiality of the findings. RF and DT correctly predicted the locations of all game events in Experiment 1 with MSEs of 9.5 and 10.6. The purpose of the previous experiment was to fine-tune when novice players deviate from professional player behaviour and establish a solid commencement for battles. Based on this discrepancy, the system provided the player with reliable recommendations on which quadrant they should be in and which event/objective they should complete. This has shown to be a beneficial method for modelling player behaviour in future research.

Keywords: Regression · Random Forest · Decision Tree · Massively Multiplayer Online Battle Arena · League of Legends

1 Introduction

League of Legends (LoL) is a team-based strategy game in which two teams of five players compete to eliminate their opponents’ bases. [4]. The game is played on a map with three lanes that players occupy according to their responsibilities. Before accessing the base, each lane contains three turrets that must be destroyed. Between the three lanes are camps of monsters that players

* This work was undertaken in the Distributed Multimedia CoE at Rhodes University.

may conquer; some camps provide a bonus to the person who kills them, while others reward money. Three jungle team objectives (Drakes, Rift Herald, and Baron Nashor) grant major benefits to the capturing team. As the game advances, players earn money by demolishing towers, killing opposing champions¹, and killing minions². Gold may be spent to purchase goods that provide an edge over the opposing side³.

Recognising the actions taken by experienced players in the aftermath of major events and providing advice on how novice players can do better is critical for capturing their attention in the gaming community. Human behaviour may be analysed with the use of machine learning, which can also be utilised to create gaming training apps.

This study use machine learning to examine how low-rank players depart from professional players in order to design a system that may assist rookie players in increasing their skills. The primary contributions address this broad research assertion by:

- using robust machine learning models trained on data gathered during and after matches. Despite carrying large quantities of relevant information, this in-game data has not yet been used in the academic literature.
- Analysing game data to learn how amateur and professional players differ.
- Based on expert player data gleaned from the best games, suggest movement patterns using visual cues.

This paper is structured as follows. The literature review is covered in Section 2, while Section 3 describes the methodology used. The experiments that were carried out and their outcomes are described in 4, and the conclusion and future work are in Section 5.

2 Literature Review

Previous research has investigated how experienced players perform differently than inexperienced players and what they do differently to achieve victory [4, 6, 14], how machine learning can be used to predict the outcomes of multiplayer online battle arena (MOBA) games [13, 1], how to create tutoring agents for new players [11], and predicting player skill learning[2]. League of Legends match timeline data has not yet been used to train machine-learning models that can predict what pro players will do during matches.

Cavadenti et al. [3] monitored player positions throughout a Defense of the Ancients 2 match (DotA2). Player traces included POI and purchases. The CHARM algorithm, for discovering frequent closed itemsets in a transaction database, performed regular itemset mining on player traces to identify aberrant behaviour.

¹ Champions are the characters that players use.

² Creatures that move up each lane and attack enemy minions, champions and tower.

³ <https://www.leagueoflegends.com/en-gb/how-to-play/>

Experiments were carried out in both quantitative and qualitative formats. The reference model was created for the quantitative tests utilising the traces of any player playing the hero Invoker⁴. The analysed traces were collected from the most active Invoker player. The majority of the patterns encompassed regular behaviours, whereas a minor fraction indicated aberrant behaviours.

The qualitative trials employed the same traces as the quantitative ones. Traces, characterised by things bought, were made in 135 games (69 defeats and 66 wins). With a minimum frequency criterion of $\theta = 1\%$, the CHARM algorithm was applied to the generated database. It was discovered that the itemsets utilised in 60% of the games were roughly equally represented in both victories and losses.

Drachen et al. [4] investigated how experienced DotA2 players travelled about the map in contrast to rookie players. Coordinate-based behaviour was divided into three categories: zone modifications, team member distribution, and time-series clustering. The first behaviour was assessed by observing how players moved across the map. The objective was to observe whether experienced players travelled about the map more. A one-way ANOVA was used to determine the average team distance during the course of a match. According to the data, experienced gamers roamed across the map more than novices.

The second behaviour was explored in order to determine the team dispersion over the map, which might be used to quantify teamwork. A one-way ANOVA was performed again to determine the average team distance across matches. Experienced players were found to be more consistent with their team spread and to have a smaller team spread.

The third behaviour experimented with unsupervised learning and time-series clustering of the mean distance between players each second. The goal was to locate matches where players had comparable movement patterns as well as variables that may cause movement patterns to diverge. The permutation distribution was employed in the research to determine the similarity between time series. In these studies, experienced players had a much lower average team distance in shorter games. The disparities in trends across skill levels were less apparent as game durations grew.

Nascimento Junior et al. [6] profiled successful and failed team behaviours in LoL matches. To choose the most useful data from all of the matches, a variety of studies were performed, including data cleaning, near-zero variance analysis, outlier analysis, data transformation, normalisation, and redundancy analysis. The converted data was placed into a K-means clustering algorithm to identify behavioural patterns and group comparable teams. Exploratory data analysis was used to characterise each cluster. The findings revealed that certain team behaviours led to victory, identifying successful and failed team behaviours.

Aung et al. [2] investigated the correlation between early learning rates and the long-term success of League of Legends players. The data was collected directly from Riot Games, and it is possible to access the same data via the Riot Games API. The data were preprocessed and analysed in a Python environment

⁴ Invoker is a playable character in DotA2

using the Pandas, Numpy, and SciPy programmes. The ML framework was built using the Scikit-learn package.

Wang [13] used Decision Tree (DT) and Logistic Regression to predict LoL game results. Both models were proven to be effective in forecasting match outcomes based on in-game characteristics such as first tower, first Dragon, first Baron, and first Rift Herald, as well as player performance. Although the DT and logistic regression are efficient, the research found that a model capable of higher generalisation, such as Random Forest (RT), would need to be adopted to forecast outcomes utilising champion selection information.

AdaBoost is a technique for enhancing classification or regression models that involves running numerous weak ML models on various training data distributions and merging the results [12]. AdaBoost combines numerous weak machine learning models into a single strong machine learning model. However, it is susceptible to noisy data [1].

RF and AdaBoost were employed in a study by Ani et al. [1] to increase the accuracy of predicting LoL match results using data acquired before and during the match.

3 Methodology

The method was broken down into three stages: data collection and processing, training using professional player match data, and event prediction using low-rank player data and feedback. Figure 1 depicts the suggested technique.

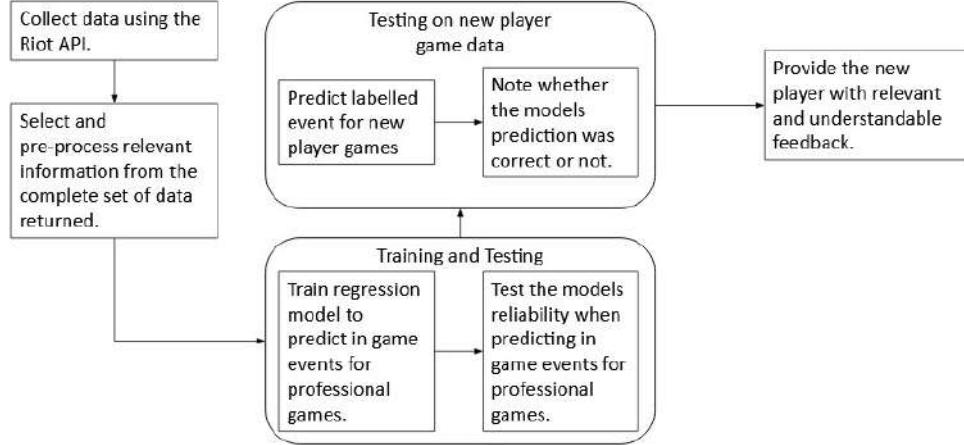


Fig. 1: High-level overview of the proposed approach

3.1 Dataset Collection and Construction

The data was gathered by *SELECTING* the top 20 rated matches of the world-renowned professional player known as Doublelift and utilising the *timeline_by_match()* function from the Riotwatcher Python module. The coding process is shown in the Listing 1.1.

```

1     api_key = "RGAPI-XXXX"
2     watcher = LolWatcher(api_key)
3     region_v4 = "NA1"
4     region_v5 = "AMERICAS"
5
6     player = watcher.summoner.by_name(region_v4, 'Doublelift')
7
8     matches = watcher.match.matchlist_by_puuid(region_v5, player['puuid'],
9         count = 20)

```

Listing 1.1: Collecting match IDs from a specific player using riotwatcher.

The method returned a dictionary containing each match's events. The selected events from dictionaries include:

- **CHAMPION_KILL**; event data for when a champion is killed.
- **CHAMPION_SPECIAL_KILL**; event data for when a special kill occurs (double kill, first blood, etc.)
- **TURRET_PLATE_DESTROYED**; event data for when a tower plate⁵ is taken.
- **ELITE_MONSTER_KILL**; event data for when an elite monster is killed.
- **BUILDING_KILL**; event data for when a building is killed (tower, inhibitor, nexus).

The event data was further analysed to acquire the following attributes:

- **timestamp**; the time that the event occurred.
- **killerId**; the ID of the player who performed the event.
- **killerTeam**; the team of the player who performed the event.
- **type**; the event type.
- **killType**; the sub-event type (the building type, the ID of the player that was killed, the type of elite monster, etc.)
- **x coordinate**; where the event occurred.
- **y coordinate**; where the event occurred.

The data were encoded for modelling by converting categorical data to integers, and the *x* and *y* coordinates were arranged into 49 quadrants so that regression models could predict quadrants as opposed to the precise *x* and *y* coordinates. This was done because coordinates are seldom identical, even for identical events; e.g., a 1920 x 1080 screen monitor has around 14,000 *x* and *y* coordinates, making the total number of potential *x* and *y* coordinate combinations 196,000,000.

⁵ Tower plating are armoured plates that towers have before 14 minutes of game time has passed. After 14 minutes, the plating falls away.

3.2 New Player Training System

A system was created to determine the magnitude of the deviation. If a player departed from the pros by more than a quadrant, a flag was raised. The model maintained track of the deviations caused by players staying in their lane for the whole of the match. After analysing the occurrences, the algorithm determined the proportion of deviations caused by the player remaining in their lane. If 50% of deviations were the result of the player staying in their lane, a message was shown to notify the player.

4 Experimental Setup

Initially, four models were trained: linear regression (LR), decision tree (DT), random forest (RF), and support vector machine (SVM) (SVM). The models were assessed using 5-fold cross-validation and AdaBoost regression throughout training. R^2 , MSE, and MAE were used to evaluate the models' dependability.

As shown in Listing 1.2, the Scikit API [7] was used to develop and train the four selected regression models using AdaBoost for a better fit. According to recommended practices, the default settings for the regression models were used.

```

1      def support_vector_machine(X, y, scores):
2          svm = AdaBoostRegressor(SVR(gamma = 'auto', kernel='rbf'), n_estimators
3          =300, random_state=42)
4          print("Support Vector Machine:")
5          cross_validation(svm, X, y, scores)
6
7          def decision_tree(X, y, scores):
8              dt = AdaBoostRegressor(DecisionTreeRegressor(random_state = 42),
9              n_estimators=300, random_state=42)
10             cross_validation(dt, X, y, scores)
11
12             def random_forest(X, y, scores):
13                 rf = AdaBoostRegressor(RandomForestRegressor(max_depth = 2,
14                 random_state = 42), n_estimators=300, random_state=42)
15                 cross_validation(rf, X, y, scores)
16
17             def linear_regression(X, y, scores):
18                 lr = AdaBoostRegressor(LinearRegression(), n_estimators=300,
19                 random_state=42)
19                 cross_validation(lr, X, y, scores)
```

Listing 1.2: Model Setup.

The interested reader should reference this article and refer to this a [Github page](#) for a complete functional system.

4.1 Experiment 1

Experiment 1 used regression models to predict the quadrant of an occurrence. The training was completed using data from 19 of the 20 games, with the final game serving as the test. The Five-Fold cross-validation findings are shown in

Table 1, and the final test results for this experiment are shown in Table 2. MSE indicates that the DT best fits the data among the four models. Even if the DT best matches the data, the MAE indicates that it also makes substantial predictions that deviate from the data. R2 provides little information outside the fact that RF generalises training data best.

Table 1: Results of Five-Fold Cross-Validation During Training.

Model	R ²	MSE	MAE
DT	0.099 (0.078)	8.855 (0.291)	159.145 (10.618)
RF	0.198 (0.015)	10.057 (0.469)	142.325 (11.352)
SVM	0.146 (0.019)	10.388 (0.325)	151.249 (7.443)
LR	0.148 (0.014)	10.432 (0.427)	150.993 (9.820)

Table 2: Results from Experiment 1 on the test set that is unseen by the model.

Model	R ²	MSE	MAE
DT	-0.436	10.600	211.316
RF	0.051	9.537	139.680
SVM	-0.150	10.393	169.311
LR	-0.341	11.256	197.321

4.2 Experiment 2

Experiment 2 used the models trained on data from individual roles. The results of the 5-fold cross-validation and testing for each role indicated that data reduction improved the dependability in the majority of instances. The tables 3 and 4 illustrate the results achieved while training and testing on top lane data, respectively. These outcomes demonstrated that DT is either very reliable or unreliable. RF generated the most reliable findings, indicating that it recognises broad patterns in the data better than the other three models. The number of models was decreased to DT and RF once these findings were made.

4.3 Experiment 3

Experiment 3 examined whether the DT and RF models were adequate for forecasting the occurrences of a single player using data from a single participant in a match.

Table 5 and 6 provide the results of testing DT and RF using data from the top laners on the Blue and Red teams, respectively. These data demonstrate how the DT may operate very well or terribly, whereas RF performs consistently well.

Graphs representing the real event's quadrant and model fitting line were generated. Figure 2 shows the plots generated by training the models for this

Table 3: Cross-Validation Scores for Top Lane.

Model	R ²	MSE	MAE
DT	-0.100 (0.199)	9.444 (0.536)	177.326 (15.797)
RF	0.189 (0.139)	9.203 (0.395)	131.666 (18.310)
SVM	0.134 (0.124)	9.879 (0.701)	141.205 (20.928)
LR	0.164 (0.148)	9.462 (0.567)	135.676 (19.513)

Table 4: Test Scores for Top Lane.

Model	R ²	MSE	MAE
DT	0.319	6.000	52.500
RF	0.117	6.107	68.066
SVM	0.015	6.884	75.910
LR	-0.063	7.512	81.956

Table 5: Test Scores for Blue Top Laner.

Model	R ²	MSE	MAE
RF	0.211	8.094	87.465
DT	0.847	2.600	17.000

Table 6: Test Scores for Red Top Laner.

Model	R ²	MSE	MAE
RF	0.103	4.786	38.189
DT	-2.946	9.429	168.000

experiment. This demonstrates that DT and RF can accurately forecast the actions of a single player throughout a match. Hence, they can determine where novice players differ from professionals. After evaluating each role, it was determined that the models accurately anticipated the happenings of the top lane.

4.4 Experiment 4

Experiment 4 evaluated the DT and RF models using data from low-rank player matches. In prior tests, it was shown that the models could accurately anticipate events when evaluated on data from professional players; hence, when tested on data from low-rank players, they could identify the areas in which the new players depart from pros. Figure 3 draws visual cues for the predictions of DT and RF based on new player data from the top lane of the blue team. The models reveal how a professional player would have moved throughout a game, whereas the actual occurrences reveal how a novice player deviates from this – thus suggesting a move.

Figure 4 depicts the movements of the new player used in this experiment and the predictions for professional movements made by the DT.

Furthermore, the new player spent the majority of their time in the top lane, but a professional player would have spent far more time assisting the rest of the

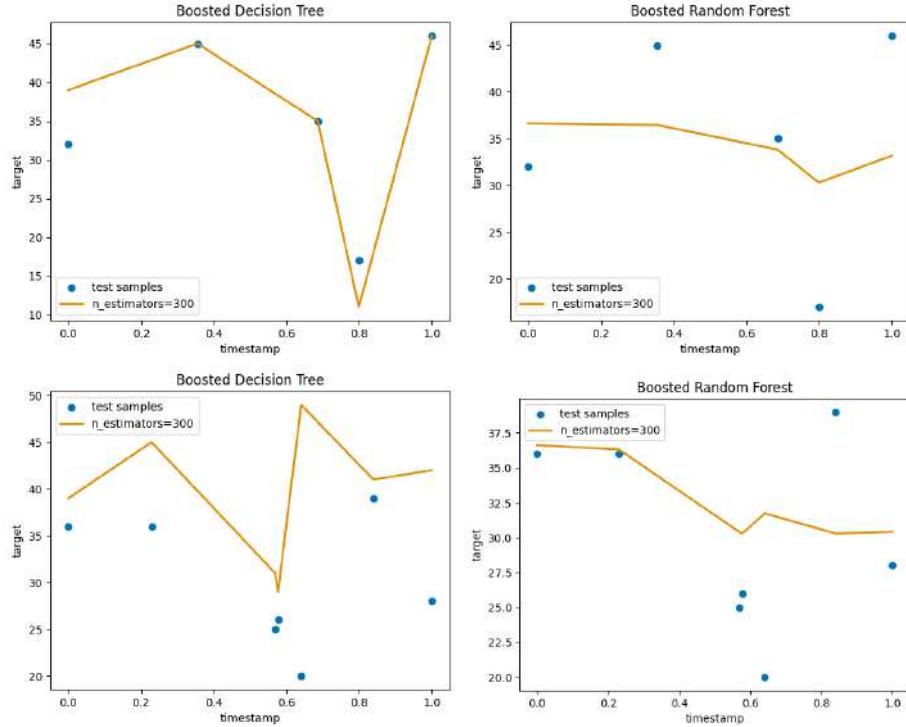


Fig. 2: Top two figures: the results from training DT (left) and RF (right) on the blue team top laners game data. Bottom two figures: the results from training DT (left) and RF (right) on the red team top laners game data.

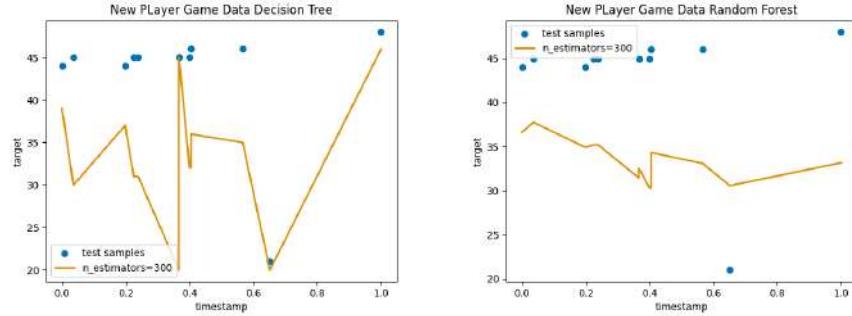


Fig. 3: Predictions of DT (Left) and RF (Right) on New Player Game Data

team by travelling across the map. This is only one example of how the system may provide gamers with ideas. Moreover, the system may be adapted for other positions, and observations could be collected about their movement habits.



Fig. 4: A depiction of the new player movements (Blue) vs the prediction for a professional player (Red). The predictions were made by the DT.

5 Concluding Remarks

This study studied how machine learning can anticipate League of Legends in-game events. The most successful machine learning regression models for the data used in this study were DT and RF. They successfully forecasted where LoL game events would occur. Using the predictions and comparing them to actual events may illustrate how the performance of inexperienced players varies from that of experts. This strategy may be utilised to identify areas in which new players can improve their in-game decision-making and help new players learn to play more like pros by emphasising their differences. Training on 19 of the 20 games with RF and DT regression produced encouraging improvements. On the unseen game, RF and DT predicted all game event locations with mean squared errors (MSE) of 9.5 and 10.6, respectively. The system provided players with continual recommendations in the form of visual cues based on the difference between novice and expert player data, as it accurately anticipated the best quadrants players should occupy and the event/goal they should pursue. This modelling approach for player behaviour can prove valuable for further studies.

In addition, recommendations for future research include the acquisition of a larger dataset of professional player statistics from a greater number of top-ranked games so as to:

- a recurrent neural network (RNN) and other time-series techniques for improved pattern recognition in diverse gaming tournaments [5, 8, 10, 9, 15].
- Drachen et al. [4] describes unsupervised computer learning approaches for recognising common patterns in time-series data. Without ground-truth training data, unsupervised methods may also aid in identifying plausible explanations for deviations from overall patterns.
- to offer additional capabilities to the new player training system and help in the determination of player movements with greater precision. Potentially, the technology might be included in the game by injecting a customisable overlay that delivers real-time feedback to novice or inexperienced players.

References

1. Ani, R., Harikumar, V., Devan, A.K., Deepa, O.: Victory prediction in league of legends using feature selection and ensemble methods. In: 2019 International Conference on Intelligent Computing and Control Systems (ICCS), pp. 74–77, IEEE (2019)
2. Aung, M., Bonometti, V., Drachen, A., Cowling, P., Kokkinakis, A.V., Yoder, C., Wade, A.: Predicting skill learning in a large, longitudinal moba dataset. In: 2018 IEEE conference on computational intelligence and games (CIG), pp. 1–7, IEEE (2018)
3. Cavadenti, O., Codocedo, V., Boulicaut, J.F., Kaytoue, M.: What did i do wrong in my moba game?: Mining patterns discriminating deviant behaviours. In: International Conference on Data Science and Advanced Analytics (2016)
4. Drachen, A., Yancey, M., Maguire, J., Chu, D., Wang, I.Y., Mahlmann, T., Schubert, M., Klabajan, D.: Skill-based differences in spatio-temporal team behaviour in defence of the ancients 2 (dota 2). In: 2014 IEEE Games Media Entertainment, pp. 1–8, IEEE (2014)
5. Hewamalage, H., Bergmeir, C., Bandara, K.: Recurrent neural networks for time series forecasting: Current status and future directions. International Journal of Forecasting **37**(1), 388–427 (2021)
6. Nascimento Junior, F.F.d., Melo, A.S.d.C., da Costa, I.B., Marinho, L.B.: Profiling successful team behaviors in league of legends. In: Proceedings of the 23rd Brazilian Symposium on Multimedia and the Web, pp. 261–268 (2017)
7. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
8. Petneházi, G.: Recurrent neural networks for time series forecasting. arXiv preprint arXiv:1901.00069 (2019)
9. Shen, G., Tan, Q., Zhang, H., Zeng, P., Xu, J.: Deep learning with gated recurrent unit networks for financial sequence predictions. Procedia computer science **131**, 895–903 (2018)

10. Silva, A.L.C., Pappa, G.L., Chaimowicz, L.: Continuous outcome prediction of league of legends competitive matches using recurrent neural networks. In: SBC-Proceedings of SBCGames, pp. 2179–2259 (2018)
11. Silva, V.d.N., Chaimowicz, L.: A tutor agent for moba games. arXiv preprint arXiv:1706.02832 (2017)
12. Solomatine, D.P., Shrestha, D.L.: Adaboost. rt: a boosting algorithm for regression problems. In: 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541), vol. 2, pp. 1163–1168, IEEE (2004)
13. Wang, T.: Predictive analysis on esports games: A case study on league of legends (lol) esports tournaments (2018)
14. Xia, B., Wang, H., Zhou, R.: What contributes to success in moba games? an empirical study of defense of the ancients 2. Games and Culture **14**(5), 498–522 (2019)
15. Yu, Y., Si, X., Hu, C., Zhang, J.: A review of recurrent neural networks: Lstm cells and network architectures. Neural computation **31**(7), 1235–1270 (2019)

Darknet Traffic Detection Using Histogram-Based Gradient Boosting *

Dane Brown¹[0000-0001-7395-7370] and Chikondi Sepula¹

Rhodes University, Drosty Rd, Grahamstown, 6140, South Africa

d.brown@ru.ac.za

[https:](https://)

[//www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/](http://www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/)

Abstract. The network security sector has observed a rise in severe attacks emanating from the darknet or encrypted networks in recent years. Network intrusion detection systems (NIDS) capable of detecting darknet or encrypted traffic must be developed to increase system security. Machine learning algorithms can effectively detect darknet activities when trained on encrypted and conventional network data. However, the performance of the system may be influenced, among other things, by the choice of machine learning models, data preparation techniques, and feature selection methodologies.

The histogram-based gradient boosting strategy known as categorical boosting (CatBoost) was tested to see how well it could find darknet traffic. The performance of the model was examined using feature selection strategies such as correlation coefficient, variance threshold, SelectKBest, and recursive feature removal (RFE). Following the categorization of traffic as "darknet" or "regular," a multi-class classification was used to determine the software application associated with the traffic. Further study was carried out on well-known machine learning methods such as random forests (RF), decision trees (DT), linear support vector classifier (SVC Linear), and long-short term memory (LST) (LSTM). The proposed model achieved good results with 98.51% binary classification accuracy and 88% multiclass classification accuracy.

Keywords: Darknet · Boosting · Classification · Feature Selection · Network Security

1 Introduction

Hackers that conduct destructive attacks use the darknet or encrypted networks since communication on these networks is anonymous. In contrast to traditional network traffic routing, in which sender identities such as IP addresses and port numbers are public, the darknet utilises protocols that conceal user data, including identities, location, preferences, and browsing behaviours. Tor and VPN are standard darknet routing protocols (VPN). [1] This darknet anonymity feature

* This work was undertaken in the Distributed Multimedia CoE at Rhodes University.

makes it more difficult for security professionals to track down cybercriminals. Consequently, preventing darknet traffic from accessing a network system is an effective and proactive method for repelling darknet-based attacks.

The most common way to find darknet traffic is to use a database with signatures of known darknet activity. If the signature of the incoming network traffic matches one of the known signatures in the database, a darknet flag is set and access is limited. When there is darknet traffic whose signature is not in the database or has not yet been identified, no darknet flag is given, and hazardous darknet traffic is allowed access.

Since machine learning models are built using normal network data to find traffic that is different from the norm, they may be able to find both known and unknown darknet traffic. However, the performance of these systems remains an important area of research, and the network security community continues to advocate machine learning techniques for performance development. Literature indicates that boosting algorithms perform pretty well in network data classification [13, 10, 3, 4, 9, 5]. CatBoost is used in this study to classify darknet traffic. A recent darknet dataset, CIC-Darknet 2020, was used to evaluate the model. Other models such as RF, DT, SVC Linear, and LSTM were examined to assess the comparative performance of the proposed model.

The premise is that histogram-based gradient boosting ensembles may provide more accurate classifications of darknet traffic than well-known machine learning methods.

The following are the most significant contributions and aims attained by this research.

1. A histogram-based gradient boosting approach for identifying darknet traffic is evaluated.
2. The performance of a darknet traffic classifier is tested using feature selection algorithms.
3. A comparison of darknet traffic classification algorithms such as boosting, bagging, distance-based, and deep learning.
4. Analysing how the synthetic minority over-sampling approach (SMOTE) affects the classification of minority classes in the darknet dataset.

2 Literature Review

Several machine-learning algorithms for categorising network traffic have been proposed. Deep learning models supplement traditional machine learning models such as distance-based, bagging, and boosting. This article is about darknet traffic.

Bakhareva et al. [3] determined the classification effectiveness of CatBoost, SVC Linear, light gradient-boosting machine (LightGBM), and logistic regression models on the CICIDS2017 network dataset. The traffic was originally classified as "normal" or "malicious," and then by the kind of assault, which included bruteforce, webattack, bot, infiltration, portscan, and denial-of-service (DoS) attack. Data samples were randomly split into 50% train and 50%

test sets. Model parameters were changed using grid search and 5-fold cross-validation. The CatBoost and LightGBM classifiers were near-perfect: 99.98% and 99.96%, respectively. Precision scores were lower for both linear SVC and logistic regression.

Extreme gradient boosting (XGBOOST) was suggested by **Bansal and Kaur [4]** as an effective means of network data categorization. The Bakhareva et al. [3] attack utilised the same CICIDS2017. The suggested model was assessed and compared to MLP, k-nearest neighbour (KNN), naive bayes, and adaptive boosting (AdaBoost). Initially, the data were preprocessed by giving numerical values to class labels. Normalized sample values ranged from 0 to 1. With a score of 99.54%, the suggested XGBOOST model has the best accuracy.

A machine learning-based NIDS system with NSL-KDD network training was developed by **Khafajeh [9]**. The system was evaluated using the DT, RF, AdaBoost, KNN, SVC Linear, logistic regression, and MLP algorithms. Attacks included probe, root-to-local (R2L), denial-of-service (DoS), and unauthorized-to-root (U2R). Model parameters were adjusted to increase performance. The most effective technique was LightGBM's 98.30% accuracy approach.

A deep learning darknet traffic detection system was created using CIC-Darknet 2020 data. **Arash et al. [2]** updated and integrated the ISCXVPN2016 and ISCXTor2017 databases to create the former data. This CIC-Darknet 2020 dataset was shown to be highly representative of the real world and is thus used in this research. Significant properties were identified and chosen prior to feeding the data to the deep learning classifier. While multi-class classification identified the traffic application, binary classification classified the data as "normal" or "darknet." The application classes included surfing, email, file sharing, P2P, chat, streaming audio and video, and voice-over-internet (VOIP). Accuracy, recall, precision, and f1-score for binary classification were all 86%.

Deep learning was used by **Sarwar et al. [13]** to detect darknet activity on the same CIC-Darknet 2020 darknet dataset. They contrasted DT with RF. Missing, redundant, and undefinable data sample rows were eliminated during data cleaning. SMOTE was utilized to resolved class imbalances for the dataset as disparities in classes samples were large. 20 important characteristics were retrieved using PCA for classification, and the model was modified to perform better. The model has 89% f1-score, 95% accuracy, 90% precision, and 88% recall.

The CIC-Darknet 2020 dataset was again evaluated on darknet traffic. **Gupta et al. [7]** produced a detection system trained using XGBOOST. Traffic types included standard, Tor, and VPN. contrasting the naive bayes, LSTM, KNN, and RF classifiers. The data were scaled from 0 to 1 and undefined and infinite values were changed to -1. Correlation coefficients eliminated extraneous traits. The model's accuracy was 98%.

In spite of the fact that boosting algorithms outperform other common machine learning models, they have not been exhaustively examined for nighttime traffic detection. Due to this deficiency, this research studies CatBoost ensemble for darknet traffic identification further.

3 The Proposed System

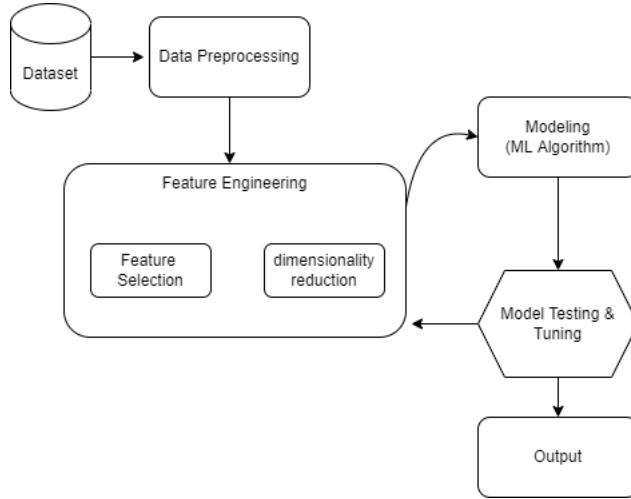


Fig. 1: The darknet traffic detection system design.

Based on the standard design in the literature [6, 14, 12, 11], the proposed darknet traffic detection system has three primary stages: data preprocessing, feature engineering, and model testing and tuning. Data preparation consists of cleansing, formatting, sampling, and scaling data. Following this is feature engineering, which removes redundant and useless features in order to choose important traits that improve model performance. Use was made of correlation coefficient, variance threshold, exploratory data analysis (EDA), SelectKBest, and RFE feature selection techniques. Figure 1 illustrates the proposed technique for detecting darknet traffic.

3.1 Data Preprocessing

The Canadian Institute of Cyber-Security website¹ was used to get the CIC-Darknet 2020 dataset. It has two target classes, 79 characteristics, and 158,659 samples in total. Non-Tor, Tor, Non-VPN, and VPN are the four labels for the first target class. Audio-Streaming, Browsing, Chat, Email, File-Transfer, Video-Streaming, P2P, and VOIP are the eight application labels for the second target class.

Data Cleaning: The darknet dataset had 47 samples with missing, duplicate, or undefined values, much as typical network data. These samples were

¹ <https://www.unb.ca/cic/datasets/darknet2020.html>-Accessed 23 October 2022

excluded because their percentage is small; hence, their removal would have no effect on the training of the model.

In Listing 1.1, the code to detect and discard samples with missing or undefined values is provided.

```
1  is_na_cols = df_2.columns[df_2.isna().sum() > 0]
2  df_2 = df_2.dropna()
```

Listing 1.1: Dropping samples with "nan" values.

Data Formatting: For binary classification, the four classes in the first target variable were reduced to two classes. The labels "Non-Tor" and "Non-VPN" have been changed to "Normal", while the labels "Tor" and "VPN" have been changed to "Darknet." Several misspelt class names in the second target class were rectified.

Listing 1.2 provides the code required to delete the four columns.

```
1  df_3 = df_2.copy()
2
3  del df_3["Flow ID"]
4  del df_3["Timestamp"]
5  del df_3["Src IP"]
6  del df_3["Dst IP"]
```

Listing 1.2: Deleting irrelevant columns from the dataset.

For binary classification, the labels were coded as "Benign" for "Non-Tor" and "Non-VPN" traffic and "Darknet" for "VPN" and "Tor" traffic as shown in Listing 1.3.

```
1  df_2["Label"].loc[(df_2["Label"] == "Non-Tor") | (df_2["Label"] == "NonVPN")] =
   = "Benign"
2  df_2["Label"].loc[(df_2["Label"] == "Tor") | (df_2["Label"] == "VPN")] = "
   Darknet"
```

Listing 1.3: Reducing the target class labels to binary.

```
1  df_3["Label"] = df_3["Label"].map({"Benign":1,"Darknet":0})
2
3  df_3["Label.1"] = df_3["Label.1"].map({"Audio-Streaming":1,"Browsing":2,"Chat":
   :3,
   "Email":4,"File-Transfer":5,"P2P":6,"Video-Streaming":7,"VOIP":8})
```

Listing 1.4: Target class label encoding.

Data Sampling: Random samples of the data were divided into 80 percent training and 20 percent testing. A substantial amount of data is given to the training split in order to develop a model that generalises effectively to unknown data.

Data Scaling: Different scales of feature values had little effect on the majority of models, notably the tree-based ones assessed in this research. MinMaxScaler of Scikit, which is identical to Data Normalisation, was used to scale the sample values for improved data analysis and display.

3.2 Feature Selection

The effect of feature engineering on model performance was investigated using a variety of feature selection strategies. Initially, redundant characteristics were deleted based on a correlation coefficient criterion of 0.80. When two characteristics have a correlation coefficient close to 1, they have the same effect on the target class and are thus redundant. Using a variance threshold of 0.00001, a collection of weak features was discarded. Close to zero variance features consist of constant values that cannot determine or aid in predicting the target class.

The code for these eliminated features based on the obtained correlation coefficients and variance is shown in the following Listing 1.5.

```

1 corr_matrix = df_train.corr().abs()
2 thresh = 0.8
3
4 upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape), k=1).astype(bool)
5 to_drop_corr = [column for column in upper.columns if any(upper[column]>thresh
6 )
7 var_thr = VarianceThreshold(threshold = 0.00001)
8 var_thr.fit(X_train_mms)
9 var_thr.get_support()
10 to_drop_thresh = [column for column in X_train_mms.columns if column not in
11 X_train_mms.columns[var_thr.get_support()]]
12 X_train_d = X_train_mms.drop(to_drop_corr, axis=1)
13 X_train_d = X_train_d.drop(to_drop_thresh, axis=1)
14 X_test_d = X_test_mms.drop(to_drop_corr, axis=1)
15 X_test_d = X_test_d.drop(to_drop_thresh, axis=1)
```

Listing 1.5: Dropping redundant and weak features based on correlation coefficient and variance threshold.

The data was plotted using EDA techniques like as violinplots, swarmplots, and heatmaps in order to identify and further minimise features that were determined to be redundant and ineffective at discriminating target groups. As shown in the Table 1, 20 important features were ultimately selected for the categorising procedure:

To determine which method of feature selection provides the most accurate model, SelectKBest and RFE were applied to choose 20 features independently. SelectKBest was used to choose the best features from the list produced by an ANOVA function, and the RF estimator was implemented in the RFE.

3.3 Model Training and Tuning

Since tuning many parameters of the suggested model did not enhance performance, the default values were kept. According to the CatBoost docs, the model's parameters may be adjusted automatically, thus tweaking isn't necessary to get good performance.

3.4 Hardware and Software

The machine learning models for the system were developed and deployed using the following platform. The system was built using a locally sourced desktop

Table 1: Features selected based on correlation coefficient, variance, and EDA techniques.

No.	Selected Features
1	Src Port
2	Dst Port
3	Protocol
4	Flow Duration
5	Fwd Packet Length Max
6	Fwd Packet Length Min
7	Bwd Packet Length Min
8	Bwd Packet Length Mean
9	Flow Packets/s
10	Flow IAT Mean
11	Fwd IAT Std
12	Bwd IAT Std
13	Fwd PSH Flags
14	FIN Flag Count
15	RST Flag Count
16	PSH Flag Count
17	Subflow Fwd Packets
18	FWD Init Win Bytes
19	Idle Mean
20	Idle Std

PC with CPU: Intel(R) Core(TM) 3.60GHz 6 Cores, RAM: 16GB, and GPU: NVIDIA GeForce GTX 750 Ti.

The packages and requirements for the utilised software platform were as follows: Microsoft Windows 10 Education OS, Python 3.10.6, Tensorflow 2.6.3, Sklearn 1.0.2, and Keras 2.6.0)

4 Experimentation and Results

Four experiments were carried out in order to test the hypothesis and achieve the study's objectives. The system's performance was evaluated using accuracy, precision, recall, f1-score, and model training time. When there are no class imbalances, accuracy, which assesses the overall accuracy of the system, is primarily utilised in binary classification. However, during multi-class classification, only precision, recall, and f1-score were recorded since accuracy was unable to offer a reflective score due to class imbalances in the second target class. The computational performance of the system was assessed by collecting the training time.

4.1 Experiment 1:

The purpose of this experiment was to see how feature selection influences model performance. First, the model was trained on all 79 attributes, and the training time and accuracy were recorded. Second, the model was trained on the 20 important characteristics, and the same metrics were gathered. Training the model on the input data prior to feature selection resulted in a slightly better accuracy score of 98.70% compared to 98.51%, as shown in Table 2. Training the model with reduced input data dimensions, on the other hand, took 41s to 61s less time. As a result, feature selection improved the model's computational efficiency, allowing it to be used in circumstances where system computing is critical.

Table 2: CatBoost accuracy scores and training time on different input data dimensions.

No. of Features	Accuracy (%)	Training Time (s)
79 features	98.70	61
20 features	98.51	41

4.2 Experiment 2

The second experiment's objective was to assess the three feature selection approaches used in this research. The model was trained and evaluated independently using the input features picked by each of the three feature selection procedures. The initial set of characteristics was chosen based on the correlation coefficient, variance threshold, and EDAs. The remaining two sets were chosen using SelectKBest and RFE. The following accuracy scores were reported in Table 3:

Table 3: CatBoost model accuracy scores based on the three feature selection techniques.

Feature Selection Technique	Accuracy (%)
EDA-based	98.45
SelectKBest	98.25
RFE	98.51

With an accuracy score of 98.51%, the RFE approach surpassed the other two feature selection procedures. This might be because RFE applies an estimate

on a set of features in order to calculate a significance score and exclude weak features. The univariate approach of seleckbest, on the other hand, examines the relevance of individual traits.

4.3 Experiment 3

The hypothesis of this research was evaluated in Experiment 3. With the exception of LSTM, which does not need feature selection, the proposed CatBoost algorithm and other well-known machine learning algorithms were assessed using 20 carefully chosen characteristics. As demonstrated in Table 4, the CatBoost model surpassed all other models with a 98.51% accuracy rate, as expected. The benefit of this algorithm is that it is an ensemble, a collection of weak learners that operate sequentially, each increasing the performance of the preceding learner [8]. The second high-performing algorithm is an RF algorithm that is also an ensemble. SVC Linear had the lowest accuracy score of 85.94 percent. This low performance might be attributable to the nonlinearity of darknet data. SVC Linear is a linear model that might fail to fit well on non-linear data, resulting to poor generalisation on unknown data.

As its tree structure is more basic than that of an ensemble, which is a collection of trees, the DT model needed the least amount of time to train.

Table 4: Accuracy scores and training time for all the models.

Model	Train Accuracy (%)	Test Accuracy (%)	Train Time (s)
DT	97.87	98.04	1
RF	98.22	98.34	21
SVC Linear	68.54	85.94	20
LSTM	95.67	96.85	180
CatBoost	98.33	98.51	30

4.4 Experiment 4

In this experiment, traffic was characterised and the application linked with the traffic was discovered. As a result of class disparities, SMOTE was implemented to oversample the minority classes. Figure ?? illustrates the distribution of data among the minority classes of the second target class. This chart demonstrates that the class disparities are large and, if not corrected, will lead to biased findings. When categorising data, the model tends to choose or favour the majority classes.

Before and after the use of SMOTE, the CatBoost model was examined to determine its influence on model prediction. The table below displays the accuracy, recall, and f1-score performance values of the model before to the

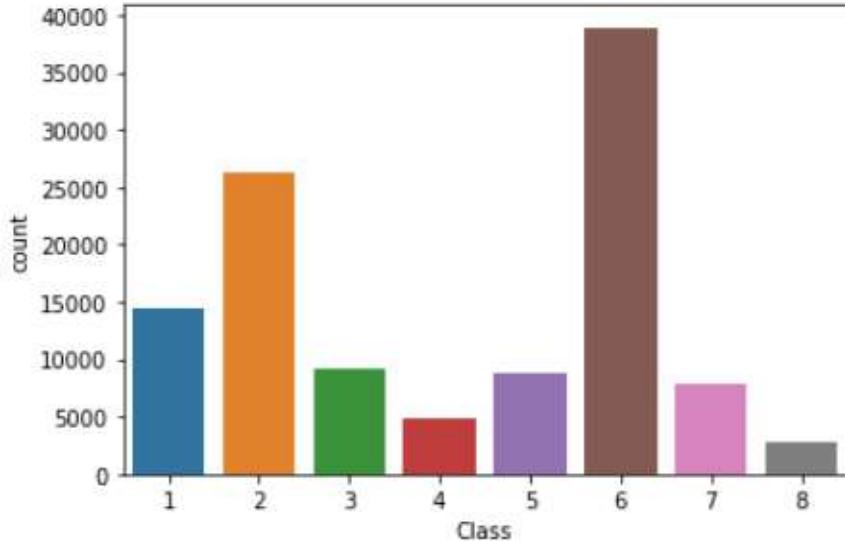


Fig. 2: Second target class data distributions of the darknet traffic before SMOTE.

application of SMOTE. The model achieved an overall accuracy of 89.99%, while individual accuracies are shown in Table 5.

Table 5: Traffic characterization scores before SMOTE.

Class (1-8)	Precision (%)	Recall (%)	F1-score (%)
Audio-Streaming	91.15	88.10	89.60
Browsing	95.96	89.15	92.43
Chat	84.59	72.31	77.97
Email	51.92	79.03	62.67
File-Transfer	79.44	91.84	85.19
Video-Streaming	97.96	97.62	97.79
P2P	69.76	80.70	74.83
VOIP	53.90	58.46	56.09

The model was assessed again after applying SMOTE, and performance scores were collected, as shown in Table 6 below, with an overall accuracy of 87.96%.

Table 6: Traffic characterization scores after SMOTE.

Class (1-8)	Precision (%)	Recall (%)	F1-score (%)
Audio-Streaming	85.92	93.08	89.35
Browsing	92.73	90.74	91.72
Chat	63.52	91.47	74.97
Email	61.37	68.33	64.66
File-Transfer	83.72	83.18	83.45
Video-Streaming	96.83	98.26	97.54
P2P	81.98	72.16	76.76
VOIP	89.08	41.21	56.35

The improvement in accuracy and f1-score of almost all minority groups is a major finding from the two tables shown above. Precision has evolved tremendously, making it very sensitive to class disparities. Prior to the oversampling of minority classes, the model was biassed towards classes with more samples in the dataset, hence the accuracy score for majority classes declined as predicted. With the use of SMOTE, the total accuracy decreases, proving that accuracy is not an acceptable measure for data with class imbalances. The initial accuracy score was high, however it did not accurately represent the model's performance. The model categorised Video-Streaming application-related traffic with good accuracy, recall, and f1-score.

In multi-class classification, the model's total accuracy of 87.96% is greater than the 86% attained by Arash et al. [2] in a comparable investigation. Similarly, the suggested model's 98.51% accuracy for binary classification beats previous experiments presented in this study [13, 7, 2] supporting the hypothesis.

5 Discussion

Multiple feature selection strategies were used to increase system performance. To assess the system, four tests were carried out. The first experiment measured system accuracy and training duration to see whether feature selection approaches were necessary. The second experiment compared the system performance differences between the feature selection approaches used. In the third experiment, several machine learning algorithms, including the suggested Cat-Boost, were assessed. The previous experiment used multi-class classification to classify network traffic depending on the linked traffic application. Due to class imbalances, however, an oversampling strategy was applied to avoid biassed findings.

The suggested machine learning method scored the best in terms of accuracy. When it came to categorising darknet traffic, the DT and RF algorithms performed marginally worse than the former. All of the tree-based models in our study required the least amount of training time. The distance-based and

deep learning models, on the other hand, achieved poor accuracy scores with the longest training period. It was also revealed that the feature selection approaches used in this study increased system efficiency but not accuracy. Finally, despite applying oversampling to minority classes to correct class imbalances, the model still produced poor performance in minority classes, which should be examined further in the future.

6 Concluding Remarks

The CIC-Darknet 2020 dataset was used to evaluate a darknet traffic identification system created utilising a histogram-based gradient boosting algorithm. There were three main stages to the system: data preparation, feature engineering, and model testing and fine-tuning. The following findings have been derived from the many experiments conducted as part of this investigation: 1) Feature selection improved the system's computational performance by requiring less time to train. 2) When compared to SelectKBest and other univariate procedures, the RFE feature selection strategy delivers better results for the model. In terms of detecting darknet traffic, the CatBoost model outperformed bagging, tree-based, linear, and deep learning models. 4) Tree-based algorithms have comparable performance ratings in darknet traffic detection. 5) While SMOTE improves model classification, the darknet dataset's large class imbalances impair model performance. Still, improvements are noticed in the unrepresented classes.

References

1. Abu Al-Haija, Q., Krichen, M., Abu Elhaija, W.: Machine-learning-based darknet traffic detection system for IoT applications. *Electronics* **11**(4), 556 (2022)
2. Arash, Habibi, L., Gurdip, K., Abir, R.: Didarknet: A contemporary approach to detect and characterize the darknet traffic using deep image learning (2020)
3. Bakhareva, N., Shukhman, A., Matveev, A., Polezhaev, P., Ushakov, Y., Legashev, L.: Attack detection in enterprise networks by machine learning methods. In: 2019 international Russian automation conference (RusAutoCon), pp. 1–6, IEEE (2019)
4. Bansal, A., Kaur, S.: Extreme gradient boosting based tuning for classification in intrusion detection systems. In: International conference on advances in computing and data sciences, pp. 372–380, Springer (2018)
5. Bhati, B.S., Chugh, G., Al-Turjman, F., Bhati, N.S.: An improved ensemble based intrusion detection technique using XGBOOST. *Transactions on Emerging Telecommunications Technologies* **32**(6), e4076 (2021)
6. Chindove, H., Brown, D.: Adaptive network intrusion detection using optimised machine learning models (2021)
7. Gupta, N., Jindal, V., Bedi, P.: Encrypted traffic classification using extreme gradient boosting algorithm. In: International Conference on Innovative Computing and Communications, pp. 225–232, Springer (2022)
8. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y.: Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems* **30** (2017)
9. Khafajeh, H.: An efficient intrusion detection approach using light gradient boosting. *Journal of Theoretical and Applied Information Technology* **98**(5), 825–835 (2020)
10. Kumar, S., Vranken, H., van Dijk, J., Hamalainen, T.: Deep in the dark: A novel threat detection system using darknet traffic. In: 2019 IEEE International Conference on Big Data (Big Data), pp. 4273–4279, IEEE (2019)
11. Le Jeune, L., Goedemé, T., Mentens, N.: Machine learning for misuse-based network intrusion detection: overview, unified evaluation and feature choice comparison framework. *Ieee Access* (2021)
12. Mohan, L., Jain, S., Suyal, P., Kumar, A.: Data mining classification techniques for intrusion detection system. In: 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), pp. 351–355, IEEE (2020)
13. Sarwar, M.B., Hanif, M.K., Talib, R., Younas, M., Sarwar, M.U.: Darkdetect: Darknet traffic detection and categorization using modified convolution-long short-term memory. *IEEE Access* **9**, 113705–113713 (2021)
14. Seniaray, S., Jindal, R.: Machine learning-based network intrusion detection system. In: Computer Networks and Inventive Communication Technologies, pp. 175–187, Springer (2022)

Efficient Plant Disease Detection and Classification for Android *

Dane Brown¹[0000-0001-7395-7370] and Sifisokuhle Mazibuko¹

Rhodes University, Drosty Rd, Grahamstown, 6140, South Africa

d.brown@ru.ac.za

[https:](https://www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/)

//www.ru.ac.za/computerscience/people/academicstaff/drdanebrown/

Abstract. This paper investigates the feasibility of using a CNN model to diagnose plant diseases in the wild. Plant diseases are a major risk to ecosystems, human and animal health, and the quality of life overall. They may reduce farm productivity drastically, leaving farmers with financial losses and food insecurity. Small-scale farmers and producers can't pay for an expert to look at their plants for plant diseases because it would cost too much. A mobile solution is thus built for the Android platform that utilises a unified deep learning model to diagnose plant diseases and provide farmers with treatment information. The literature-recommended CNN architectures were first analysed on the PlantVillage dataset, and the best-performing model was trained for integration into the application. While training on the tomato subset of the PlantVillage dataset, the VGG16 and InceptionV3 networks achieved a higher F1 score of 94.49% than the MobileNetsV3Large and EfficientNetB0 networks (without parameter tuning). The VGG model achieved 94.43% accuracy and 0.24 loss on the RGB PlantVillage dataset, outperforming the segmented and greyscaled datasets, and was therefore chosen for use in the application. When tested on complex data collected in the wild, the VGG16 model trained on the RGB dataset yielded an accuracy of 63.02%. Thus, this research revealed the discrepancy between simple and real-world data, as well as the viability of present methodologies for future research.

Keywords: Android · CNN Architecture · Classification · Deep Learning · Plant Disease

1 Introduction

Climate change influences food supply chains and increases the incidence of plant disease, which has detrimental effects on the surrounding environment and results in lower yields, therefore harming the economy, human health, and ecosystems [6].

As a result of agricultural practises such as monoculture and the production of pathogen-vulnerable plants, the destructiveness of plant diseases increases.

* This work was undertaken in the Distributed Multimedia CoE at Rhodes University.

Increase their rate of dissemination in accordance with contemporary worldwide commerce [9]. Misdiagnosis of plant diseases may result in a number of unfavourable results, including major effects on human health and the surrounding environment if pesticides are carelessly sprayed in excessive amounts. It also increases costs for farmers due to crop losses and inadequate disease control techniques [24].

Plant disease monitoring systems are also required to monitor pathogens throughout a vast territory in order to reduce the risk that a pathogen would cause significant crop damage, thus reducing the infection's impact on productivity, ecosystems, and humans [21]. The Program for Monitoring Emerging Diseases (ProMED) is a global disease monitoring organisation that studies illness occurrences to anticipate prospective epidemics in order to limit disease transmission and serve as a warning system for people.

The use of Convolutional Neural Networks (CNNs) for autonomous plant disease detection, as noted by [23], makes disease detection easier, cheaper, and more accessible to farmers. Farmers now have more access to information, enabling them to conduct crop loss prevention strategies and provide more data to disease monitoring organisations.

CNNs are employed because they are robust and efficient when processing large image collections. [18]. In this research, different CNN architectures and settings for the datasets are looked at to find the best architecture and settings for the built system. The PlantVillage dataset is used to train models during testing and to make the final model for the system. It contains 54,305 images of 24 diseases on 14 different plant species.

The optimum model can identify and categorise plant diseases in a mobile situation with limited resources by evaluating alternative models.

Contributions were made as guided by these objectives:

1. A labelled dataset of plant leaves, including both healthy leaves and leaves with disease signs. Implement several image processing algorithms for input images utilised by the final model for detection.
2. To select an architecture for future research, compare several CNN architectures on a subset of the plant image dataset using the appropriate statistic.
3. Compare the results of the various dataset settings for the selected CNN architecture.
4. Evaluate the model's performance using real-world data to see whether it can be applied to new data with complex histories.

2 Literature Review

Mohanty et al. [16] has studied the use of deep learning in a smartphone application to diagnose and classify agricultural disease. Due to improvements in smartphone processing speeds and cameras, widespread smartphone use makes deep learning accessible to a broader range of consumers.

CNNs outperform other deep learning and machine learning techniques, such as Recurrent Neural Networks (RNNs) and Support Vector Machines (SVMs),

for plant disease diagnosis [14]. CNNs are time-consuming to train, but they can classify images rather rapidly, making them suitable for use in Android apps [16]. Contrary to CNNs, machine learning approaches need data preprocessing and feature extraction prior to model training, but CNN designs just necessitate complex preparation.

Machine learning algorithms perform better when the data is more organised. To improve the performance of models trained on a limited structured dataset, it might be supplemented. Fang et al. [7] created radar profiles with four classifications using a Deep Convolutional Generative Adversarial Network (DCGAN).

As explained by Salimans et al. [22], the DCGAN utilises convolutional and deconvolutional layers in their discriminator and generator, respectively, incorporating the advantages of CNN feature extraction and therefore enhancing image analysis abilities. Moreover, it serves as a form of data augmentation that can allow classifiers to train on more varied data for better inference in the wild.

Fang et al. [7] trained a CNN model with both actual and DCGAN samples, and subsequently trained another model using only genuine cases. The model trained using a mixed dataset increased in accuracy, converging at 89.37% after 30 epochs, according to the researchers.

Because noise may significantly affect the look of a leaf in an image, picture preprocessing reduces noise and enhances detection accuracy [17]. To accommodate for harsh lighting conditions, we may apply the Retinex algorithm [11] to enhance the image.

Iniyam et al. [14] examined the feasibility of edge detection and blob detection for feature extraction using artificial neural network (ANN) and SVM models. These strategies somewhat enhanced the ANN and SVMs accuracy. In the identification of plant diseases, SVMs were unable to beat neural networks such as convolutional or recurrent neural networks, even with feature extraction.

Patel et al. [19] used ImageNet weights to examine alternative colourspace transformation algorithms on four CNN architectures: VGG, InceptionV3, ResNet, and DenseNet. The images were RGB, HSI, and other colourspace transformed using the CINIC dataset [2]. For all models, the CIVIC and SVHN RGB colourspace datasets fared the best, with the least loss and the highest accuracy. These results support the conclusion reached by [16] that the RGB dataset delivers the highest degree of accuracy. This drop in performance might be attributed to data loss in the greyscaled dataset.

Poole and Brown [20] explored the use of VIS-IR imaging and if it enables deep learning algorithms to classify plant dehydration stress with much more accuracy than visible imaging. They found that VIS-IR imaging was significantly better at detecting plant stresses, but that visible imagery can also be successful with sufficient data and effective augmentation strategies.

De Silva and Brown [3] collected images of 10 species of agricultural plants in their natural environments for disease categorization. The images were captured using a Canon EOS 700D camera equipped with a Kolari Vision Hot Mirror filter for RGB photographs and a Kolari Vision K665 filter for NIR images.

ResNet-50V2 outperformed other deep-learning models on both the VIS and NIR datasets, with test accuracy rates of 98.35 and 94.01 percent, respectively. Examining eight CNN models with train-test splits ranging from 10:90 to 90:10 using the same NIR dataset [5].

Aditya et al. [1] created a sequential model using the Keras API. They trained the model from scratch using a dataset of 56,725 pictures, without considering whether the model was modest and efficient enough for a mobile application. An accuracy rating of 96.84% was achieved using batch normalisation to enhance learning and a regularisation tool to minimise overfitting data using the Keras image processing module that builds batches of tensors from the dataset.

Francis and Deisy [8]’s CNN architecture comprised of four convolutional layers activated by ReLU and two fully connected layers activated by Softmax. The objective was to create a small model for mobile and embedded applications. The 44Kb model has an accuracy of 88.7%. The smallest model was trained by the authors.

Mohanty et al. [16], unlike Francis and Deisy [8], used ImageNet-pretrained AlexNet and GoogleNet architectures and trained models from scratch. The accuracy of the RGB image **PlantVillage** dataset was 99.34% when using the GoogleNet transfer learning model. With transfer learning and RGB data, CNNs performed best.

Hassan et al. [12] compared the CNN architectures InceptionResnetV2, InceptionV3, MobileNetV2, and EfficientNetB0. EfficientNetB0 outperformed the other three models in terms of accuracy, recall, and f_1 score, with 99.56% accuracy. It had fewer parameters and took less time to train than the other models, but it needed considerable parameter tweaking, which was outside the scope of this research.

Goncharov et al. [10] investigated the applicability of a model trained on the PlantVillage dataset in a real-world setting. This research looks at the VGG16, InceptionV3, and InceptionResNetV2 models, which were pre-trained on imageNet data, then trained on PlantVillage data, and finally assessed on real-world pictures. Goncharov et al. obtained the results shown in Table 1 for the aforementioned models; the low accuracy was attributed to background noise, the presence of leaves on an image as opposed to having a leaf as in the PlantVillage data, and the low resolution of the real-world images used for testing.

Table 1: Goncharov et al.’s PlantVillage Trained Models Tested on Real-World Data

Architecture	Accuracy
VGG16	19.73%
InceptionV3	30.78%
InceptionResNetV2	39.87%

3 Methodology

The proposed end-to-end system can be summarised as: XML code that generates the front-end visual aspects designed in incision studiom while Kotlin code runs the interactions between different aspects and pages that call the machine learning model. The interested reader is referred to [Github page](#) and encouraged to reference this paper.

3.1 Dataset Collection

Hughes and Salathé [13] compiled the PlantVillage dataset of plant diseases, and Mohanty et al. [16] curated the PlantVillage dataset used in this work. The photographs depict a single plant leaf set on a white/light blue background. The collection consists of 54 305 photographs organised into three sets: RGB, segmented, and greyscale.

De Silva and Brown [4] collected a real-world dataset consisting of pictures of plant leaves in a complex natural ecosystem without being separated from the rest of the plant, as is the case with the PlantVillage dataset. This dataset is much closer to data in the wild and can be used to show the gap between modelling simple and real-world data.

3.2 Image processing

To handle the application's diverse lighting conditions, the Retinex image enhancing technology was used. This application employs the single-scale Retinex algorithm. It is based on centre-surround Retinex by Jobson et al. [15].. It is the difference in logarithms between the image at (x, y) and its center-surround average at (x, y) (x, y) . The Gaussian distribution is used to calculate the center-surround average of an image at certain locations (x, y) .

3.3 Image Segmentation

The image segmentation technique removes the background of the leaf image, leaving just the leaf and disease spots. A green, yellow, and brown mask is constructed using an HSV image of the plant's leaf and the `bitwise_or` operation. The Retinex algorithm was used to minimise image noise and improve identification in challenging circumstances. The approach fails when the background of a picture is green, yellow, or brown because it assumes it is a leaf.

3.4 Dataset Augmentation

The dataset's different classes were sampled unequally, and several dataset augmentation procedures were tested to reduce the inaccuracy induced by unequal sampling.

The TensorFlow ImageDataGenerator Keras class is used to rotate and flip images vertically and horizontally. This class is used to distinguish the photographs in each sample class, enabling the model to train on images that are more dissimilar.

A Deep Convolutional Neural Network (DCGAN) was considered to augment the dataset in order to compensate for unequal sampling across classes and to improve the dataset of photographs from the natural world. The DCGAN's discriminator and generator networks use batch normalisation¹ layers and the ADAM². Optimise using the values from Table 2.

Table 2: ADAM Optimiser Parameters

<i>learning rate</i>	0.001
<i>beta_1</i>	0.5

The discriminator computes loss using BinaryCrossentropy. As shown in Listing 1.1, the total loss for the model is derived by adding the model's real and fictitious losses. The real loss is retrieved from an image in the training dataset, while the fake loss is derived from the image that was produced. The loss evaluates the discriminator's ability to discern between actual and artificial images.

```

1 def discriminator_loss(real_output, fake_output):
2     real_loss = cross_entropy(tf.ones_like(real_output), real_output)
3     fake_loss = cross_entropy(tf.zeros_like(fake_output), fake_output)
4     total_loss = real_loss + fake_loss
5     return total_loss

```

Listing 1.1: Discriminator Loss.

The Conv2DTranspose operation and the LeakyReLU activation function are used by each of the generator's six hidden layers to scale the input image to 256x256 pixels. The generator's output layer constructs a tensor of outputs using the Conv2D operation and tanh activation. The loss evaluates the generator's deceit by comparing the discriminator's judgements against the real output.

```

1 def generator_loss(fake_output):
2     return cross_entropy(tf.ones_like(fake_output), fake_output)

```

Listing 1.2: Generator Loss.

3.5 Convolutional Neural Network Architecture

CNN performs feature extraction and classification with high accuracy without prior data processing or feature engineering. CNN architectures considered were VGG16, InceptionV3, EfficientNetsB0, and MobileNetV3Large. According

¹ Normalises weights for faster training.

² Adaptive Moment Estimation is a backpropagation approach for efficient gradient descent.

to the literature, these are the best structures for categorising plant pictures in a mobile context. Among the architectures tested, the VGG16 and InceptionV3 models received the highest scores for accuracy, precision, and f1. Because it was designed to perform well under severe constraints and a constrained computational budget [25], the VGG architecture uses less RAM and has fewer parameters than the tested Inception network.

4 Disease Detection and Classification Back-end System

Figure 1 depicts a diagram with components of the machine learning backend system for identifying and categorising plant diseases. The objectives direct the system design based on Section 1 and the methodologies concepts explained above.

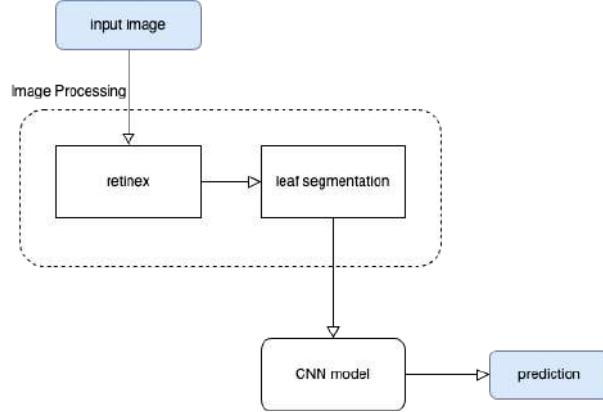


Fig. 1: Plant disease detection and identification system: a high-level view.

The TensorFlow Keras library is utilised to construct the models that will be used to analyse the different dataset configurations and the efficacy of converting the dataset using the TensorFlow ImageDataGenerator class to enhance model performance. Additionally, the Keras library is used to construct the model for the final system and to conduct testing using real-world data.

Image preparation procedures are performed on the final system input in order to make the input photos to be as similar as possible to the images that are used for prediction in the wild. This adds extra variation in order to improve accuracy on unseen data and complex backgrounds.

5 Experiments conducted

Each CNN architecture described in Section 3.5 was modelled. For the final model selection, further testing with a subset of the dataset was undertaken

to assess model performance and compare the models to one another. Model fine-tuning was not as robust as indicated by the literature.

The following is an overview of each of the experiments conducted:

- Using the findings of this experiment, choose a model for further study based on the performance of the model on the tomato dataset, which has 10 classes.
- Evaluate the efficiency of changing the dataset using the TensorFlow ImageDataGenerator class in order to enhance model performance.
- Examine the different dataset combinations to see which combination produces the highest performance.
- After picking a model, its performance on the whole dataset must be assessed. item A small dataset including four classes was collected from the real world, and the chosen model will be trained and assessed using this dataset.

6 Results and Discussion

6.1 Model Testing on Tomato Dataset

Each design was tested using a subset of the PlantVillage dataset, which contained nine leaf diseases and nine healthy tomato species. The dataset consists of 16011 photographs separated into three sets: training, validation, and testing (60:20:20).

All models are trained using transfer learning with ImageNet weights and a *0.001* learning rate. The batch size in this experiment was ten, each model was trained for ten epochs, and the prediction time is the average of ten tests.

According to Table 3, MobileNetV3Large and EfficientNetB0 require less memory, making them ideal for usage in a low-resource mobile context. However, they have the lowest f1 score, suggesting that they are not as exact and robust as VGG16 and InceptionV3, offsetting their benefit of using less memory.

Table 3: Architecture Testing

Architecture	Memory Taken	f1 score	Time to Predict	Accuracy
MobileNetV3Large	33.6mb	3.4	0.051sec	86%
EfficientNetB0	20mb	0.0	0.047sec	20%
VGG16	367.7mb	94.49	0.041sec	94%
InceptionV3	994.5mb	91.49	0.060sec	91%

The MobileNetV3Large model has poor precision and recall, resulting in a low f1 score, but good accuracy, suggesting that the model is underfitting due to the unequal sampling across classes in the dataset used in this experiment.

Validation losses for the EfficientNetB0, InceptionV3, and VGG16 models converge at 2.2, 0.3, and 0.5, respectively, with the InceptionV3 and VGG16 models having somewhat higher validation losses than the training loss, indicating that they are underfitting the training data. The losses of the MobileNetV3Large model had not stabilised by the ninth epoch, and the training loss was much more than the validation loss. This demonstrates that MobileNetV3Large does not generalise well and needs additional training epochs and training data to reach convergence and decrease losses.

Among the models tested, the VGG16 and InceptionV3 models performed the best. They have higher f1 scores and lower training and validation losses, suggesting that they are more robust models with lower error rates. EfficientNetB0 and MobileNetV3Large have substantial losses, suggesting that they create erroneous output, which explains their extremely low f1 scores; hence, they are not robust models and need further data and parameter change to improve. They were not further investigated since parameter change was beyond the scope of this inquiry. Because of their greater performance, the VGG16 and InceptionV3 models are employed for future exploration.

6.2 Comparing Datasets

The use of models trained on a single dataset are examined in this section, and the various colourespaces are assessed on images from a different dataset in order to discover a model that generalises successfully across image colourespaces. A model that performs well on both segmented and traditional RGB datasets is ideal for the application. Furthermore, the greyscale dataset is analysed for comparison purposes. To train the models, the parameters in Table 4 are utilised.

Table 4: Training Parameters For Comparing Datasets

learning rate	0.001
epochs	10
batch size	60

InceptionV3 And VGG16 Models Trained on RGB Data: The performance of models trained on the RGB dataset on images in RGB, greyscale, and segmented colour spaces was examined in this experiment.

Although their training losses are equal and stabilise at the same low point, the InceptionV3 model has a higher validation loss than the VGG16 model. The validation loss of the InceptionV3 model rises from a low point, indicating that the model overfits the training data. The validation loss of the VGG16 model does not stabilise and does not increase in the same way as the InceptionV3 model does.

Table 5: VGG16 and Inception RGB Model Test Results

Model	Test Dataset	Accuracy	Loss	Precision	Recall
VGG16	RGB	94.43%	0.24	94.43%	94.02%
	Segmented	69.53%	2.53	69.53%	68.50%
	greyscale	63.64%	2.25	65.31%	62.69%
InceptionV3	RGB	89.63%	0.87	89.90%	89.57%
	Segmented	75.44%	2.26	76.11%	75.18%
	greyscale	71.25%	3.28	71.84%	71.05%

The RGB training results are summarised in Table 5, revealing that the model performs well on images in the colourspace in which it was trained, with the models performing best on RGB images. The VGG16 model performed best, but the InceptionV3 model performed better on segmented and greyscaled images, although with a bigger loss than the VGG16 model, indicating that greyscale images created more inaccurate output. More training is needed to discover the overall trend of both models' losses in order to determine if the models have been overfitted; this test determines whether image colourspace is adequate for this application without attempting to improve model performance.

InceptionV3 And VGG16 Models Trained on Segmented Data: In this experiment, segmented dataset-trained models were assessed on RGB and greyscale photographs, in addition to segmented pictures.

The model's training losses had stabilised at a low value, which was somewhat greater than the models trained on RGB data in the previous section. As noted in the preceding section, the validation loss for the InceptionV3 model is growing, indicating that the model has been overfit to the training data. It is necessary to stabilise the validation loss of the VGG16 model, which may be achieved by training for extra epochs.

Table 6: VGG16 and Inception Segmented Model Test Results

Model	Test Dataset	Accuracy	Loss	Precision	Recall
VGG16	RGB	72.37%	2.13	73.51%	71.85%
	Segmented	93.28%	0.31	93.77%	93.07%
	greyscale	39.74%	5.87	41.31%	38.88%
InceptionV3	RGB	71.90%	3.10	72.37%	71.59%
	Segmented	89.34%	0.79	89.57%	89.23%
	greyscale	47.93%	9.07	48.44%	47.64%

When tested on photographs in the colour space in which it was taught, the VGG16 model performed the best across all criteria. It was tested with segmented photographs in this case and functioned better with RGB images. Despite a high loss rate indicating incorrect output, the InceptionV3 performed

best with greyscale images. The segmented dataset outperforms the VGG16 model.

InceptionV3 And VGG16 Models Trained on greyscale Data The performance of models trained on the greyscale dataset on images in RGB, greyscale, and segmented colour spaces was tested in this experiment.

As additional epochs are trained, the InceptionV3 validation loss continuously rises; this implies that the model may be overfitting on training data, but further training is needed to determine the overall trend of the loss curve. Both models' training losses stabilised at a modest level, as they have in previous research. In contrast to the loss seen with RGB data, the validation loss for the VGG16 model is rather stable. It is larger than the training loss, and additional training is necessary to find the long-term trend, same as the validation loss for the Inception model.

Table 7: VGG16 and Inception greyscale Model Test Results

Model	Test Dataset	Accuracy	Loss	Precision	Recall
VGG16	RGB	78.43%	1.24	79.71%	77.91%
	Segmented	59.50%	3.30	60.48%	58.97%
	greyscale	89.27%	0.46	90.29%	88.79%
InceptionV3	RGB	68.54%	3.14	69.20%	68.35%
	Segmented	40.26%	7.72	40.73%	39.90%
	greyscale	87.18%	1.08	87.47%	87.05%

Compared to its counterparts from preceding sections, the greyscaled model fared the worst across all criteria. CNN models prefer extra information, such as richer colour, to draw more features from in order to differentiate objects, therefore this decrease in performance is to be expected.

Comparing RGB Trained InceptionV3 and VGG16 Models With RGB data Further: InceptionV3 and VGG16 models are compared using the same model parameters, including a hidden layer of 1024 nodes and the Rectified Linear unit activation function, which makes the model easier to train and frequently leads to the model performing better.

The models' performances are similar, with VGG16 producing somewhat better outcomes. The training accuracy of both models starts to decrease after seven epochs, dropping from 98.5% for the VGG16 model to 98.5% and 97.7% for the InceptionV3 model, respectively. According to Table 3 shows that the VGG16 model uses less storage space and makes predictions at comparable rates when trained with the tomato dataset.

According to Table 5, the VGG16 model demonstrated improved accuracy, precision, and recall on unseen test data, as well as a lower testing loss. On segmented and grayscale data, the performance of the Inception model is superior.

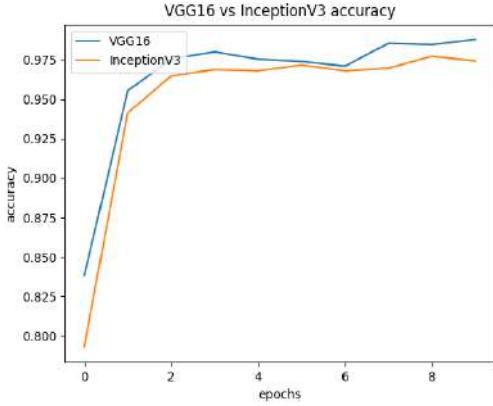


Fig. 2: VGG16 vs InceptionV3 Training Accuracy

The bulk of the model’s RGB images utilised in this application will not be separated. The VGG16 architecture was chosen for the application and future study.

Testing Model on Real-World Data: Utilizing the information in Table 8 and 9, a VGG16 model was trained for five iterations. The VGG16 model’s performance and robustness are tested using RGB PlantVillage-trained real-world plant data. Table 10 displays the test results for PlantVillage.

Table 8: Dataset Used To Train Model For Real-World Usage Testing
Dataset categories

Tomato Healthy	1591
Tomato Yellow Leaf Curl Virus	5357
Potato Healthy	152
Peach Healthy	360

Table 9: Real-world Dataset Used To Test Model
Dataset categories

Tomato Healthy	100
Tomato Yellow Leaf Curl Virus	62
Potato Healthy	74
Peach Healthy	79

Table 10: Test With PlantVillage Data

Accuracy	Loss
97.79%	1.29

Testing using PlantVillage RGB data reveals in Table 10 that the model achieves good accuracy with substantial loss. This demonstrates that the model works effectively when evaluated against the PlantVillage dataset. As the classes are not uniformly sampled, the high loss and high accuracy may potentially imply that the model overfits the data. The very low training loss of 2.4495e-05 compared to the comparatively large testing loss of 1.29 is another indication of overfitting.

Table 11: Test With Real-World Data

Accuracy	Loss
63.02%	33.93

As seen in Table 11, the real-world test results in a very big loss, indicating that the model generates considerably inaccurate output. In contrast, [10] trained a VGG16 model using PlantVillage data, achieving a 15.08% accuracy rate on unseen real-world images, resulting in a model with a rather high degree of plant recognition accuracy.

Discussion of Results The best-performing model was the VGG16, which was trained over 10 epochs using the whole PlantVillage RGB dataset. As a result, it is used across the system. Because it creates fewer wrong outputs, the VGG16 model consistently obtained the lowest loss across all datasets. Furthermore, the training loss was lower than the validation loss, showing that the model generalises well.

The programme sends the image to the backend, which is a RESTful API developed using the Flask Python framework and processes it. The Retinex and segmentation algorithms are applied on the image separately. The original and segmented photographs are then used for prediction. The prediction is based on the most confident outcome, which is then conveyed to the programme and shown to the user.

The real-world data test shows that the PlantVillage VGG16 RGB model is durable and can work with real-world data; nevertheless, more data is needed to prevent overfitting and regularise the model. This test illustrates that an application can run with high accuracy in real-world conditions while employing a VGG16 model without image processing. When evaluated with real-world data, the model's accuracy drops due to background noise and the complexity of pictures with several leaves [10].

7 Concluding Remarks

The model with the best performance is the VGG16 model, which was trained over 10 epochs using the whole PlantVillage RGB dataset. As a result, it is used across the system. Because it delivers less incorrect output, the VGG16 model consistently received the lowest loss across all datasets. Furthermore, the training loss was less than the validation loss, indicating that the model generalises well.

The picture is sent by the application to the backend, which is a RESTful API built using the Flask Python module that processes the image. The Retinex and segmentation algorithms are applied individually to the picture. Forecasting is then done using the segmented and original photos. The prediction is based on the most confident result, which is subsequently submitted to the programme and shown to the user.

The test with real-world data shows that the PlantVillage VGG16 RGB model is resilient and can operate with real-world data; however, further data is required to avoid overfitting and regularise the model. This test, which uses a VGG16 model without image processing, suggests that an application can perform properly in real-world circumstances. When compared to real-world data, the model's accuracy decreases due to background noise and the complexity of photos that include multiple leaves. This is thus a promising approach for future research dealing with plant image data in the wild.

References

1. Aditya, D., Manvitha, R., Mouli, C.R.: Detect-o-thon: Identification of infected plants by using deep learning. *Global Transitions Proceedings* **2**(2), 336–343 (2021)
2. Darlow, L.N., Crowley, E.J., Antoniou, A., Storkey, A.J.: Cinic-10 is not imagenet or cifar-10. University of Edinburgh (2018)
3. De Silva, M., Brown, D.: Plant disease detection using deep learning on natural environment images. In: 2022 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), pp. 1–5 (2022), <https://doi.org/10.1109/icABCD54961.2022.9855925>
4. De Silva, M., Brown, D.: Plant disease detection using deep learning on natural environment images. In: 2022 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), pp. 1–5, IEEE (2022)
5. De Silva, M., Brown, D.: Plant disease detection using multispectral imaging. Advanced Computing, In Press (1) (2022)
6. Fan, S.: Global food policy report. International Food Policy Research Institute (2016)
7. Fang, W., Zhang, F., Sheng, V.S., Ding, Y.: A method for improving cnn-based image recognition using dcgan. *Computers, Materials and Continua* **57**(1), 167–178 (2018)
8. Francis, M., Deisy, C.: Disease detection and classification in agricultural plants using convolutional neural networks—a visual understanding. In: 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), pp. 1063–1068, IEEE (2019)

9. Fry, W.E.: Principles of plant disease management. Academic Press (2012)
10. Goncharov, P., Ososkov, G., Nechaevskiy, A., Uzhinskiy, A., Nestsiarenko, I.: Disease detection on the plant leaves by deep learning. In: International Conference on Neuroinformatics, pp. 151–159, Springer (2018)
11. Hao, W., He, M., Ge, H., Wang, C.j., Gao, Q.W.: Retinex-like method for image enhancement in poor visibility conditions. Procedia Engineering **15**, 2798–2803 (2011)
12. Hassan, S.M., Maji, A.K., Jasiński, M., Leonowicz, Z., Jasińska, E.: Identification of plant-leaf diseases using cnn and transfer-learning approach. Electronics **10**(12), 1388 (2021)
13. Hughes, D.P., Salathé, M.: An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. CoRR **abs/1511.08060** (2015), URL <http://arxiv.org/abs/1511.08060>
14. Iniyan, S., Jebakumar, R., Mangalraj, P., Mohit, M., Nanda, A.: Plant disease identification and detection using support vector machines and artificial neural networks. In: Artificial intelligence and evolutionary computations in engineering systems, pp. 15–27, Springer (2020)
15. Jobson, D.J., Rahman, Z.u., Woodell, G.A.: A multiscale retinex for bridging the gap between color images and the human observation of scenes. IEEE Transactions on Image processing **6**(7), 965–976 (1997)
16. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers in plant science **7**, 1419 (2016)
17. Neelakantan, P.: Analyzing the best machine learning algorithm for plant disease classification. Materials Today Proceedings pp. 2214–7853 (2021)
18. Panchal, A.V., Patel, S.C., Bagyalakshmi, K., Kumar, P., Khan, I.R., Soni, M.: Image-based plant diseases detection using deep learning. Materials Today: Proceedings (2021)
19. Patel, S., Oza, U., Kumar, P.: Is deep learning image classification invariant to color space transforms. ResearchGate (2019)
20. Poole, L., Brown, D.: A multispectral and machine learning approach to early stress classification in plants (2022)
21. Ristaino, J.B., Anderson, P.K., Bebber, D.P., Brauman, K.A., Cunniffe, N.J., Fedoroff, N.V., Finegold, C., Garrett, K.A., Gilligan, C.A., Jones, C.M., et al.: The persistent threat of emerging plant disease pandemics to global food security. Proceedings of the National Academy of Sciences **118**(23), e2022239118 (2021)
22. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Advances in neural information processing systems **29** (2016)
23. Singh, V., Misra, A.K.: Detection of plant leaf diseases using image segmentation and soft computing techniques. Information processing in Agriculture **4**(1), 41–49 (2017)
24. Sun, J., Yang, Y., He, X., Wu, X.: Northern maize leaf blight detection under complex field environment based on deep learning. IEEE Access **8**, 33679–33688 (2020)
25. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826 (2016)

Text-to-Speech and Speech-to-Text Converter – Voice Assistant

Sagar Janokar¹, Soham Ratnaparkhi¹, Manas Rathi¹, and Alkesh Rathod

Department of Engineering, Sciences and Humanities (DESH) Vishwakarma Institute
of Technology
sagar.janokar@vit.edu
soham.ratnaparkhi21@vit.edu manas.rathi21@vit.edu alkesh.rathod21@vit.edu

Abstract: According to the International Agency for Prevention of Blindness's Vision, there are around 32 million people who live with avoidable blindness and a further 259 million with preventable visual impairment that is decent to acute. Though there may be a few existing solutions to this problem, none of the solutions provide an all-in-one experience like this project does. Our proposed model uses Automatic Speech Recognition (ASR) to recognize user's speech and convert it into text format. For converting text to speech, we are using google text-to-speech (gTTS) engine and Microsoft's SAPI5 which provides voice to our model. This Voice Assistant can not only be used to read a Word document or a PDF file but also search content on Google or Wikipedia.

Keywords: Voice Assistant · Read · Search Contents · Website · Documentation · Guide · Prerequisites · Download

1 Introduction

In recent times, the growth of Artificial Intelligence and Human-Machine interfaces is huge. Virtual Assistants, especially Voice Assistants have contributed to this growth immensely. These voice assistants not only aid the blind but are also a boon to specially-abled people. Voice assistants are intelligent software that can run on any device and respond to voice commands. These devices include computers gaming consoles mobile phones, smart gadgets, TV consoles, virtual reality (VR) headsets, cars, and internet of things (IoT) devices. Voice assistants can also be referred to as personal digital assistants (PDAs) or virtual assistants.

This year, 123.5 million adults in the United States will use voice assistants at least once per month, and that base will continue to grow over the course of the following several years. The number of people who use voice assistants will continue to rise, eventually reaching over half of the adult population in the United States. Even though the number of users is expected to rise at a slower rate through the conclusion of our prediction period in 2025, we anticipate that in the next three years, slightly over 48% of adults in the United States will be monthly users of this technology.

It should not come as a surprise that younger members of the millennial generation are the ones most likely to employ a voice assistant. This year, we anticipate that close to two-thirds of people in the age range of 25 to 34 will use voice assistants every month. This number falls below 50% among members of Gen X (those aged 42 to 57), and it is slightly around 30% for members of the baby boomer generation (ages 58 to 76).

However, still, this technology has some disadvantages. The method needs massive databases and hard-coding to construct words. Speech synthesis uses

more CPU. Speech is unnatural and impersonal. It's difficult to capture all conceivable words, emotions, prosody, tension, etc. Text-based pronunciation analysis is important. Perfect systems are hard to develop. Humans may have trouble filtering background noise.

Digital speech recognition technology has been embedded in personal computers since the start of this century with Microsoft, Apple, Philips, etc., continuously working on it. Many top firms use oral dialog systems for designing such system devices as Microsoft Cortana, Amazon Alexa, Alphabet's Ok Google, Apple Siri, etc. Vocal recognition, voice language apprehension, dialogue manager, natural language production, text-to-speech converter, and knowledge base are the six components of a general conversation system. However, there are some major issues with these voice assistants as either they are multifunctional but very expensive or lack the functionalities to make a cheap product. This paper proposes a Voice Assistant which is not only a multifunctional software but also a free and open-source application that can be used by anyone. Lately, a lot of research has been done on Voice Assistants and various new technologies have come up. Text-to-speech conversion using different speech syntheses has been carried out. Text-to-speech consists of two phases i.e., the first phase is text analysis and the second one is the generation of speech waveforms [1]. Few researchers integrated Image processing algorithms, OCR, and Text-to-Speech (TTS) synthesis to build a Voice Assistant[2]. The Raspberry Pi module connected to a camera was often used to capture the input image and this input image was further processed and utilized using OCR [3]. It was proposed, a software named eSpeak which read out the audio formatted file and delivered it as an output with the help of a speaker[4]. Author proposes using an Optical Character Recognition (OCR) based smart book reviewer for the visually impaired[5]. The unique method of creating a local voice assistant without using cloud services was put forth in [6].

During our research, we found that one of the technologies for natural language interactions with virtual personal assistant systems is a computing device configured to receive audio input, distort the audio input to generate several distorted audio variants, and perform speech recognition on the audio input and the distorted audio variants [7]. We came to the conclusion that more investigation into the Voice Personal Assistant, also known as a VPA, is required. A VPA is a digital assistant that uses voice recognition, natural language processing, and speech synthesis to assist users of voice recognition applications [8]. Further in our research, we found that there is a need for a system that can build voice and face recognition for student attendance systems. While this has a range of positive effects, the most essential one is that it provides lecturers with assistance in effectively monitoring the attendance of their students. In this particular method, the marking of pupils' attendance is accomplished by the use of human biometrics [9]-[10].

The automatic speech-to-text and speech-to-speech summarization methods presented in [11] is based on speech unit extraction and concatenation. A two stage summarising method using important sentence extraction and word-based sentence compaction is examined for the first scenario [13]. The speech recognition results are extracted and concatenated to create sentence and word units that optimise the weighted total of linguistic probability, amount of information, confidence measure, and grammatical likelihood of concatenated units [14]. We have a profound understanding of the reading norms of our native language, which is mostly unconscious. In elementary school, they were taught to us in a reduced manner, and we refined them year by year. However, it would be rather audacious to assert that it is

just a matter of time until the computer would likely surpass the human in this regard [15]. Despite the current level of our knowledge and skills, as well as the recent advancements made in Signal Processing and Artificial Intelligence, we must raise misgivings. In actuality, the reading process utilises the furthest, sometimes unconsidered depths of human mind [16].

2 Methodology

The Voice Assistant has been coded using Python3. Microsoft's SAPI5 and pyttsx3 library enables the Voice Assistant to speak and comprehend the speech of the user. We have used the recognize_google() method present in the recognizer class of the speech recognizer library which empowers the Voice Assistant to convert speech to text. To convert text to speech, we use the methods in pyttsx3 and win32com libraries. Fig.1 shows the flowchart for working with this Voice Assistant. Python's pyttsx3 module is capable of converting text into spoken language. In contrast to other libraries, it may be used without an internet connection and is compatible with Python versions 2 and 3. To obtain a reference to a pyttsx3.Engine instance, an application must first call the pyttsx3.init() factory method. It is a straightforward process to transform written words into spoken words with this program. The "sapi5" program for Windows is responsible for providing both the female and male voices that are supported by the pyttsx3 module. The first voice is female. It is compatible with the following TTS engines: sapi5 – SAPI5 on Windows nsss – NSSpeechSynthesizer on Mac OS X and espeak – eSpeak on every other platform. Our tool greatly focuses on 2 major spheres, i.e; Speech to Text conversion and Text to speech conversation.

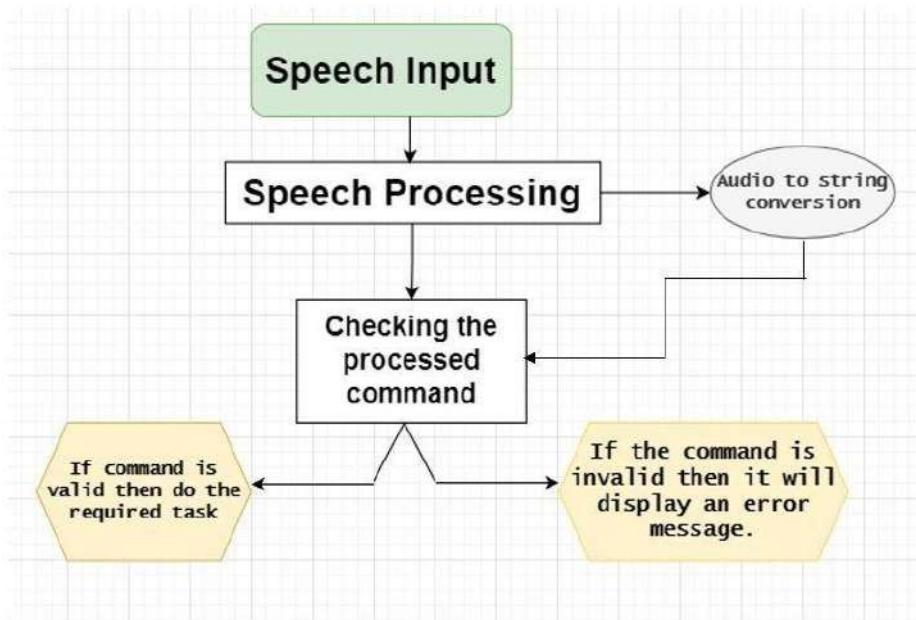


Fig.1. Flowchart of stated methodology

1.1 Speech to text Conversion

- Initially, the Voice assistant records the speech. It requires a certain threshold frequency value (300 units) for recording the voice successfully.
- It then converts the raw speech into a string datatype. This string is then assessed by the Voice Assistant.
- Once the commands are received by the Voice Assistant, it checks the commands provided by the user with the list of commands present in its database.
- If the command matches, then the Assistant will perform the desired task else it gives the corresponding error message.

The Voice Assistant generally records for 9 seconds, but this duration can be changed as per the user's requirement. The overriding theory behind the working of this AI-based Voice Assistant is Automatic Speech Recognition. Automatic Speech Recognition, abbreviated to ASR for its more common name, is the technology that enables people to use their voices to communicate with a hardware electronic system in a manner that, in its most advanced iterations, is remarkably similar to how people speak to one another in everyday life.

Our program converts speech to text operated by first playing an audio file and then producing a verbatim transcript that may be edited on the device in question. Voice recognition allows the program to do this task. Computer software separates auditory information from spoken words and transfers those signals into text using characters called Unicode relies on linguistic algorithms.

The process of converting voice to text is accomplished by the use of a sophisticated machine-learning model that is comprised of numerous stages. Words are created when a succession of sounds are expelled from someone's lips, which also results in a series of vibrations. For speech-to-text technology to function, it must first detect these vibrations and then use an analog-to-digital converter to turn them into a digital language. The sounds from an audio file are read by an analog-to-digital converter, which then analyses the waves by measuring them in high detail and filters them so that only the necessary sounds are heard. After the sounds have been split into hundredths or thousandths of seconds, the phonemes may finally be matched to them. In every given language, the unit of sound that is used to differentiate one word from another is referred to as a phoneme. For instance, the English language has something in the neighborhood of 44 phonemes. After that, the phonemes are put through a network that is based on a mathematical model, and the model compares the phonemes to well-known sentences, words, and phrases.

Natural Language Processing, often known as NLP, is at the core of the most advanced version of the ASR technologies that are currently being developed. While we still have a long way to go before reaching the peak of progress, we are already witnessing spectacular outcomes in the shape of intelligent smartphone interfaces such as Siri on the iPhone and other systems utilised in business and sophisticated technology settings. Despite the fact that there is still a considerable distance to travel before we reach the summit of progress, we are already achieving fantastic outcomes. This application of ASR comes the closest to facilitating a true conversation between humans and artificial intelligence. Despite having an average accuracy of between 95% and 98%, these NLP systems can only attain such results under perfect conditions. These ideal circumstances include that the queries presented to them by humans are basic yes-or-no questions or that there are only a limited number of viable responses depending on the specified keywords.

The basic events that cause any Automatic Speech Recognition program to work as shown in Fig.2 are as follows:

1. The audio feed is recorded.
2. Of the recorded audio file, a wave file is created of your speech.
3. Background noise is removed from the wave file, and the loudness is normalized.
4. The resulting filtered waveform is then dilapidated into phonemes. (Phonemes are the fundamental sounds that makeup language and words. There are 44 of them in English, and they are made up of sound blocks like "wh", "the", "ka", and "t".)
5. Each phoneme is like a link in a chain, and by studying them in order, you may learn more about them. Starting with the first phoneme, the ASR algorithm uses statistical probability analysis to create whole phrases and eventually whole sentences."
6. ASR as shown in Fig.2 now having "understood" your words, can respond to you in a meaningful way which in our case is being assessed by the Voice Assistant to give a meaningful response.

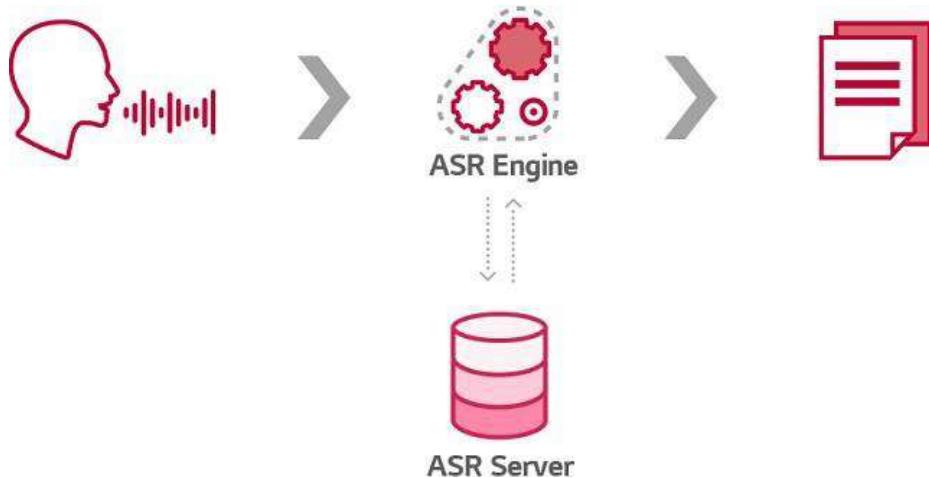


Fig.2. Working of ASR

1.2 Text to speech conversion

Deep learning research advances have made it possible for us to produce voices that sound human. Our research revolved around few terminologies which are

- Phoneme - The smallest unit of sound, or phoneme, is what distinguishes one word's pronunciation and meaning from another.
- Mel-spectrogram - It is created by reducing the dimensionality of the audio's short-time Fourier transform (STFT) by applying a non-linear treatment to the frequency axis. It highlights low-frequency characteristics that are crucial for identifying speech and downplays high-frequency information that are often noise.
- Prosody - the rhyme schemes and sound patterns utilised in poetry.

Fig3 shows the basic workflow of Text to speech conversion model

Text is the input to our model, which passes through various blocks before being translated to audio. Let's examine how each of these building elements adds to the process.

- Preprocessor
- Tokenize: Tokenize a statement into words

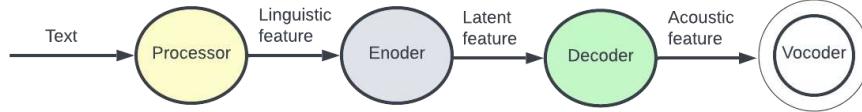


Fig.3. Workflow of TTS engine

- Phonemes/Pronunciation: It divides text input into phonemes according to their pronunciation.
- Phoneme duration: Represents the entire amount of time each phoneme in an audio recording takes up. Pitch is a crucial characteristic for conveying emotion, and it has a substantial impact on the prosody of speech.
- Energy: Indicates the magnitude of frame-level mel-spectrograms and impacts the volume and prosody of speech directly.

The Linguistic characteristic contains just phonemes. In actuality, energy, pitch, and duration are used to train the energy predictor, pitch predictor, and duration predictor, which are employed by the model to generate a more natural output.

- Encoder

The encoder accepts linguistic information (Phonemes) as input and generates an n-dimensional embedding as output. This embedding between encoder and decoder is referred to as the latent feature. Other aspects, such as speaker embedding (which will be detailed in the following blog), are concatenated with these and provided to the decoder. In addition, the latent features are utilised to forecast the audio's energy, pitch, and length, which play a significant part in determining the audio's naturalness.

- Decoder

The decoder converts information embedded in the Latent processed feature to the Acoustic feature, i.e. Mel-spectrogram.

We are using mel-spectrograms output instead of the speech/audio directly from the decoder. This is due to the fact that audio contains more variation information (such as phase) than Mel-spectrograms. This results in a greater information gap between the input and output for text-to-audio creation than for text-to-spectrogram generation. Mel-spectrograms are therefore recommended.

- Vocoder

It transforms the Acoustic characteristic (Mel-spectrogram) into waveform output (audio). It is possible to use a mathematical model, such as that of Griffin Lim, or to train a neural network to learn the mapping between melspectrogram and waveforms. In actuality, learning-based strategies typically perform better than the Griffin Lim method.

Instead of immediately predicting waveform using the decoder, we divided this hard and intricate operation into two parts, first predicting the melspectrogram from Latent processed data and then creating audio using melspectrogram.

Directed dialogue dialogues and natural language conversations make up the two primary categories of varieties seen in automatic speech recognition software. Conversations in Directed Dialogue are a simplified kind of automatic speech recognition in action. They are made up of machine interfaces that speak to you and instruct you to respond verbally with a particular word chosen from a restricted list of options. This is how they form their response to your specifically defined request. ASR may be utilized in a variety of ways, one of which is through Directed Dialogue dialogues. Directed conversation automatic speech recognition software is utilized rather frequently in a variety of customer service interfaces, including automated telephone banking systems. Natural Language Conversations are much more advanced variants of ASR. Rather than providing you with heavily restricted menus of words you are permitted to use, these variants attempt to simulate real conversation by allowing you to communicate with them in an open-ended chat format. One of the most cutting-edge instances of these kinds of technologies is the Siri interface seen on Apple's iPhone.

2 Results

After installing all the required packages and libraries, the code was implemented using an executable file. The executable file and its setup as well as the source code is present on the website of this project. The source code has been administered in Python 3.x. Underneath are a few outputs that we got from the Voice Assistant. Saying "Hello Python" will activate the Voice Assistant. Then the table in Figure 3 will be displayed on the screen. It shows the tasks and their corresponding commands that Voice Assistant can perform. Saying "close python" will deactivate the Voice Assistant system. After performing the task, the Voice Assistant (VA) will ask the user whether he/she wants to continue. Depending on the user's answer, it will decide whether to continue or not.

Fig.4 indicates the tasks that the tool can do Following is a brief description of the tasks that the proposed research does.

- Text-to-speech output: When we type out a sentence, the VA will speak it out. For this, the text is processed by using the underlying principles of ASR and NLP.
- Search on google output: As shown in Fig.5 and 6, When the user asks the voice assistant to search 'Voice Assistant', it acts by searching on google. It receives the request through google and then performs this task. For this, the user's speech is first processed into text and then this text is queried by Google. Similarly, the Voice Assistant searches the text on Wikipedia too, but the only difference here is that it speaks out and prints a few lines of 'summary' of the query to be searched on Wikipedia. It then asks the user whether he wants to open the respective webpage.

Say "Hello Python" to activate the Voice Assistant!		
Listening...		
Recognizing...		
I can do the following :-		
Sr. No.	Task	Command
1	Speak Text entered by User	text to speech
2	Search anything on Google	Search on Google
3	Search anything on Wikipedia	Search on Wikipedia
4	Read a MS Word(docx) document	Read MS Word document
5	Convert speech to text	Convert speech to text
6	Read a book(PDF)	Read a book
7	quit the program	Python close

Fig.4. Tasks that the tool can do

- Read an MS Word(.docx) or PDF file Once the respective command to do this is called, the voice assistant will ask the user the location of the document to read. In the case of reading a pdf file, VA will also read out the number of pages, author, and title of the document if possible and then give certain interactive options to the user related to reading the document. Fig.7 shows the output after calling the 'read a book command'.
- Convert speech to text Once this command is executed, the Voice Assistant acts like a dictator and prints the sentence spoken by the user. This dictation feature does a good job of converting your voice into the written word.

I can do the following :-		
Sr. No.	Task	Command
1	Speak Text entered by User	text to speech
2	Search anything on Google	Search on Google
3	Search anything on Wikipedia	Search on Wikipedia
4	Read a MS Word(docx) document	Read MS Word document
5	Convert speech to text	Convert speech to text
6	Read a book(PDF)	Read a book
7	Quit the program	Python close
 Listening... Recognizing...----- 100% 0:00:00 search on Google		
What do you want me to search on Google?		
 Listening... Recognizing...----- 100% 0:00:00 voice assistant		

Fig.5. Asking VA to search 'voice assistant' on Google

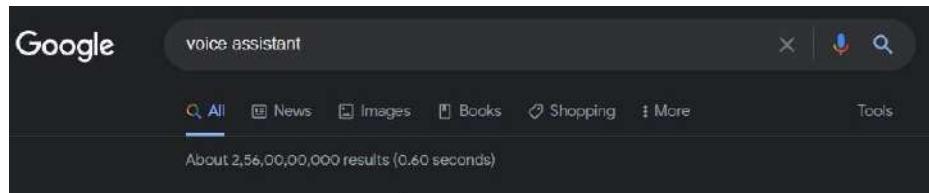


Fig.6. Search results produced by the voice assistant

```

Listening...
Recognizing... 100% 0:00:00
read a book

Enter the document's location - D:\College\Books\Friends.pdf

Book Details :-



| Sr. No. | Property | Value                                   |
|---------|----------|-----------------------------------------|
| 1       | Title    | How to Win Friends and Influence People |
| 2       | Author   | Dale Carnegie                           |
| 3       | Pages    | 263                                     |



Say 1 or "ONLY PRINT INDEX" - if you want me to print the book's index.
Say 2 if you want me to print and make me speak out the book's index.
Say any key if you don't want to print the index.

Listening...
Recognizing... 100% 0:00:00
any Ki

1. Print/speak a single page
2. Print/speak a range of pages
3. Print/speak a Lesson
4. Read/speak a whole book

Listening...
Recognizing... 100% 0:00:00
print speak a single page

Page Number - 2

```

Fig.7. Results of the 'Read a book' command

3 Conclusion

The primary purpose of this initiative is to deliver inexpensive software to underserved groups in society, such as the blind, visually impaired, and differently-abled. This project has been built using various open-source libraries and packages. It can be used to search on the web, or as a dictator and a Word document or a PDF file reader. The main factor that differentiates this Voice Assistant from others is its synergism with the user. Speech-to-text, like all other types of technology, offers several advantages that help us better the procedures we go through daily. Automatic speech recognition technology helps users save time by producing accurate transcripts in real-time, allowing for more efficient use of the technology. Using our tool, it takes minimal effort to say a few words than type them on a small smartphone screen in handsfree manner without keeping your hands occupied. This allows users to do multiple tasks at the same time. The customer experience is streamlined by utilizing natural language processing to provide convenience, accessibility, and seamlessness in the customer experience transformation.

4 Limitations and Future scope

While dictation technology is a powerful tool, it's still in its early stages, so its performance is not flawless and perfect. As it only creates literal text, you risk getting a wrong, clunky, or inaccurate transcript. Voice-to-text technology isn't perfect, so speech data must be reviewed and corrected by a human. Clear recordings with less background noise is necessary as Voice recognition software can produce a high-quality transcript only if the audio is free from background noise or accents. You must provide voice commands for punctuation because the tool doesn't recognise breakpoints and tone changes. Sound quality is intelligibility. Output sentences can have some glitches in cases where there is a confusing pronunciation of words. In future, background noise cancellation can be integrated to improve the accuracy of results.

References

1. Hay Mar Htun, Theingi Zin, Hla Myo Tun "Text to speech conversion using different speech synthesis" IJSTR 2015
2. Shalini South, Jagadish S Kallimani "OCR Based Facilitator for the Visually Challenged" 2017 International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEECCOT)
3. Polyakov E.V., Mazhanov M.S. et al. "Investigation and Development of the Intelligent Voice Assistant for the Internet of Things Using Machine Learning" 2018 4. Moscow Workshop on Electronic and Networking Technologies (MWENT)
5. Hasan U. Zaman, Saif Mahmood, Sadat Hossain, Iftekharul Islam Shovon "PythonBased Portable Virtual Text Reader"
6. Subhash H, Prajwal N Srivastava et al. "Artificial Intelligence-based Voice Assistant" 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)
7. Prabhakar Manage, Rajamani M.Kulkarni, et al. "An Intelligent Text Reader based on Python"
8. Yadav, Avanish Vijaybahadur, Sanket Saheb Verma, and Deepak Dinesh Singh."Virtual Assistant for blind people." 2021 International journal of advance scientific research and engineering trends 6, no. 5 (2021).
9. Porwal, Ritik, Ujjawal Tomar, Vishakha Dubey, Akshita Mishra, and Gourav Mandloi. "Voice Assistant." (2021).
10. Muthumari, M., V. Akash, K. Prudhvi Charan, P. Akhil, V. Deepak, and S. PhaniPraveen. "Smart and Multi-Way Attendance Tracking System Using an ImageProcessing Technique." In 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), pp. 1805-1812. IEEE, 2022.
11. S. Furui, T. Kikuchi, Y. Shinnaka and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," in IEEE Transactions on Speech and Audio Processing, vol. 12, no. 4, pp. 401-408, July 2004, doi: 10.1109/TSA.2004.828699.
12. Yi Ren, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, Tie-Yan Liu Proceedings of the 36th International Conference on Machine Learning, PMLR 97:5410-5419, 2019.
13. Allen, J., Hunnicutt, M. S., Klatt, D. H., Armstrong, R. C., Pisoni, D. B. (1987). From text to speech: The MITalk system. Cambridge University Press.
14. Ren, Yi, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. "Fastspeech: Fast, robust and controllable text to speech." Advances in Neural Information Processing Systems 32 (2019).
15. Dutoit T. High-quality text-to-speech synthesis: An overview. Journal Of Electrical And Electronics Engineering Australia. 1997 Mar;17(1):25-36.
16. Zen, Heiga, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. "Libritts: A corpus derived from LibriSpeech for text-to-speech." arXiv preprint arXiv:1904.02882 (2019).

CNN combined with FC Classifier to combat Artificial Penta-digit text-based Captcha

¹S Abhishek, ²Sanjana S, ³Mahima Chowdary Mannava, ⁴Anjali T

¹abhishekabi2002@gmail.com, ²sanjana.suresh.iyer@gmail.com,
³mannavamahima@gmail.com, ⁴anjaliraj87@gmail.com

Department of Computer Science and Engineering, Amrita School of Computing
Amrita Vishwa Vidyapeetham, Amritapuri, India.

Abstract

Captcha technology has grown in popularity along with the growth of the Internet. The Completely Automated Public Turing Test to Tell Computers and Humans Apart uses captcha technology to discriminate between humans and robots. The most common type of CAPTCHA is text-based. The majority of CAPTCHAs that use text have been broken. However, prior research has primarily relied on complex and ineffective preparation methods to attack text CAPTCHAs. This study builds a deep CNN network model that can recognize a 5-character text captcha by researching captcha recognition technology.

Keywords: **Captcha, Convolutional Layer, Fully Connected, Connectionist Temporal Classification, Long Short-Term Memory.**

I. Introduction

Scientific research today regularly makes use of deep learning network which is a great success in several fields, including detection systems, speech recognition, image identification, and natural-language processing. Deep learning's capacity to actively learn characteristics without the aid of artificial design is its main advantage over conventional pattern recognition. A test known as CAPTCHA was developed to prevent websites from being regularly and quickly visited by an autonomous program and squandering network resources. Most companies that offer online services utilize CAPTCHA tests to stop users from performing specific actions, such as filling out forms.

A collection of coarse resolution, distorted texts with character contractions and background noise—standard CAPTCHAs—must be successfully read and entered by the user into an input area. Computers find it difficult due to the noise, which makes it difficult for software to distinguish between the characters. With Convolutional Neural Networks (CNN), computers could perform these CAPTCHA tasks rapidly and correctly [1].

The development of novel, higher-security CAPTCHAs and the assessment of the robustness of existing CAPTCHA types can benefit from an easy-to-use, effective, and precise approach to identifying CAPTCHAs. The same method employed for CAPTCHA identification may be used in various fields, such as forensic investigation and license plate analysis, among many others. This work focuses on CAPTCHA, which basically mixes English characters and random digits for security purposes. It is easy to make, independent of the user's

language and culture, and challenging to break by raw force. Using standard computer languages, we can create a visual with numbers and characters. To make the characters more challenging for computers to decipher, the CAPTCHAs must analyze their twisting conglutination and add background noise [2]. This research also conducted extensive tests to evaluate the security of most text-based Captchas resistance mechanisms.

II. Related Works

In early research, pre-processing, segmentation, and recognition were the three primary phases of segmentation-based assaults. As a result of their distinctive designs and generating algorithms, several CAPTCHA schemes may have various properties. Therefore, attackers must develop different pre-processing and segmentation techniques. There is no doubt that the entire assault procedure is time-consuming and incapable of generalization. Nevertheless, the general approaches that several earlier research purported to present have certain drawbacks [3].

The image's characters are recognized using these two processes. As an illustration, Yan and Ahmad effectively segmented Microsoft CAPTCHAs and recognized them using several models, with a detection accuracy of 60%. Mori and Malik can use a form context technique to identify CAPTCHAs in photos [4].

Through character segmentation and recognition, Chellapilla and Simard are also able to crack CAPTCHAs. The segmentation approach is also used in domestic academic studies on CAPTCHA recognition. The k-nearest neighbour (KNN) technique was proposed by Wang Yang et al. to be used for identification verification codes. The digits in the modern CAPTCHAs will partially overlap, making it exceedingly challenging to divide the single character, damaging the recognition accuracy and hindering the computer from automatically identifying the CAPTCHAs and boosting network security [5].

LeCun et al. advocated using deep learning approaches to recognize handwritten digits to address the earlier issues in light of the limits of conventional image processing techniques. They must all split the photographs, though. Instead, we immediately obtain the result using the entire set of photographs as input [6][7]. Tanget et al. stated that they implemented a CNN and offered a general technique in 2018. However, most of these systems relied on time-consuming, ineffective pre-processing and segmentation techniques [8].

This research suggests a straightforward, all-purpose strategy for defeating text CAPTCHAs. This approach makes use of a CNN-based attention-based paradigm. The experimental findings showed that our model achieved excellent success rates without the need for segmentation or any other pre-processing methods. The efficiency of all popular defensive methods against deep learning assaults was also thoroughly examined.

According to the analysis, deep learning assaults may render the segmentation-resistance concept inapplicable. Several unusual CAPTCHAs were also discussed in this article to assess their security. Our methodology is general for different CAPTCHA schemes since all objective CAPTCHAs were defeated with excellent success rates.

III. Dataset

The dataset for the research of Captcha Recognition consisted of 1070 text-based CAPTCHA pictures in png format. Rodrigo Wilhelmy and Horacio Rosas provided the data (2013). The photographs contain five-letter words that may or may not contain numbers. Noise has been added to the photos (blur and a line). There are 19 characters,

with a character range of five for each dataset picture. Digits (2, 3, 4, 5, 6, 7 and 8) and certain lowercase English alphabets (b, c, d, e, f, g, m, n, p, w, x, y) were used. Except for the character n, which is used twice as frequently as other characters, each character is used almost equally frequently. They are PNG files with dimensions of 200 x 50. It would be ideal for creating OCR algorithms using this dataset. Despite being in grayscale, they nonetheless have three dimensions.

IV. Proposed System Design

A. Architecture

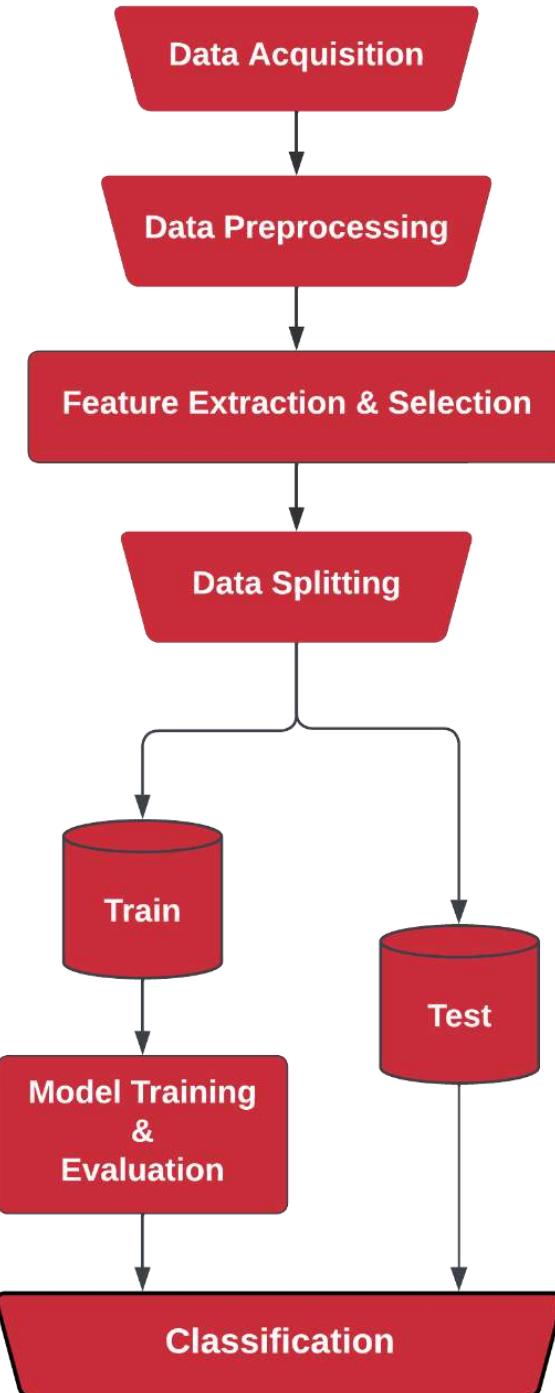


Figure 1 - Architecture

This identification model has an advantage over earlier captcha recognition techniques. It does not require pre-treatment of the original captcha picture and may perform effective captcha recognition without undergoing the time-consuming pre-treatment procedure.

The CNN network design serves as the foundation for the recognition model. The CNN design initially uses the CNN network structure to extract the characteristics of the input pictures. The five-character captcha may be recognized by the model that was built in this study. We may immediately obtain the network's outcome by using decoding techniques.

Using CRNNs (Convolutional Recurrent Neural Networks) is recommended while developing Optical Character Readers. Now that the neural network has been trained, we must utilize this matrix to decode its output and train the neural network, which entails computing its loss. Both goals are accomplished with the aid of CTC operation. The suggested system's design and workflow are shown in [Figure 1].

Data Pre-processing

Initially, the picture was read in a grey scale for the 5-letter word captcha images. These strategies are used to limit the number of properties from the initial feature set to decrease model complexity, reduce model overfitting, increase the efficiency of model computation, and decrease generalization error. The core feature extraction concept is data compression while retaining pertinent information. Following that, further pre-processing procedures were carried out:



Figure 2 – Image before Pre-Processing

Adaptive Thresholding

Adaptive thresholding techniques were used to choose a pixel's threshold depending on the dataset's surrounding portions of the picture. We got good performance for photos with a range of brightness levels because we obtained varying thresholds for distinct portions of the same image.



Figure 3 – Image after Adaptive Thresholding

Closing

The closing procedure, which includes dilation and erosion, was used to enlarge the dataset's pictures and erode the expanded versions, expanding and contracting the images to improve their clarity.



Figure 4 – Image after Closing

Dilation

Dilation was performed by running a kernel across the whole picture. The kernel region's maximum pixel value was established, and the anchor point of the kernel was modified to coincide with the value. The outcome was an enlargement of the white region in the photograph.



Figure 5 - Image after Dilation

Smoothing

Convoluting the picture with a low-pass filter kernel produces an image blur. Noise may be removed with it. This filter blurs the edges of the image by eliminating high-frequency material (such as noise and edges). There are blurring methods available that won't obscure the edges, though. While gaussian blurring is similar to average blurring, it uses a weighted mean instead than an essential mean. This means neighbourhood pixels nearer the centre pixel give the average more "weight."



Figure 6 - Image after Smoothing (Blurring)

Gaussian smoothing is used to eliminate noise that roughly corresponds to a Gaussian distribution. The ultimate effect is that, compared to the standard procedure, our image is less blurred but more "naturally blurred." We will be able to retain more of the image's edges with this weighting. The kernel for Gaussian smoothing is $M \times N$, where M and N are both odd numbers, just like for average blurring.

Partitioning

The OpenCV rectangle() function is dedicated to creating algorithms that can address issues with computer vision. With the help of the OpenCV rectangle function, a rectangular-shaped hollow box may be drawn on any picture the user provides.



Figure 7 - Image after Batch Partitioning

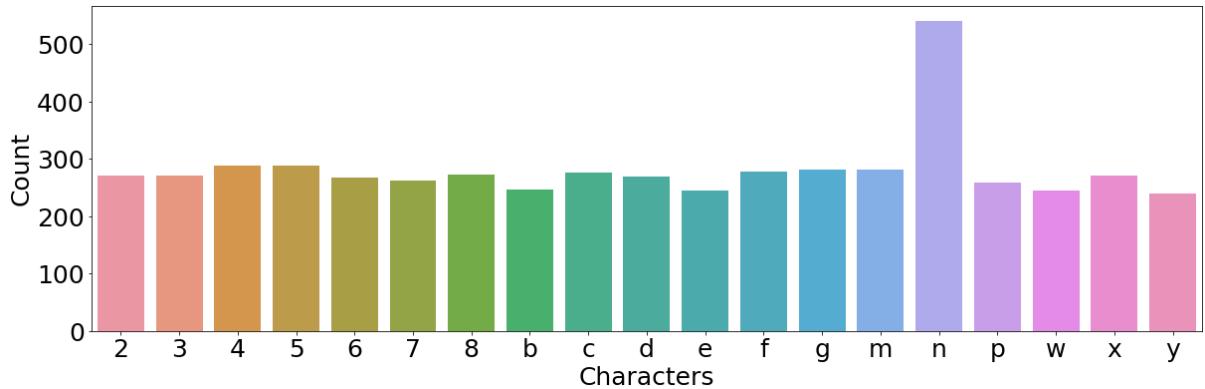


Figure 8 - Label Distribution in Captchas

More instances of this character will be contained in the test set since label n has a nearly two times higher count than any other label as shown in [Figure 8]. Now, one can see that just 19 characters—2,3,4,5,6,7,8, and b,c,d,e,f,g,m,n,p,w,x,y—are utilized in the CAPTCHA pictures. Apart from n, which is used twice as frequently as other characters, each character is used nearly equally frequently.

Feature Extraction & Selection

The goal of feature extraction is to build a new feature subspace by extracting or extracting from the original features. Due to the stringent security measures used, including the thick noise lines, varied foreground and background colours, and a wider variety of character classes used in the CAPTCHA scheme, even humans find it challenging to recognize the image. Each element of the CAPTCHA picture has a specific form, size, and orientation [9].

The essential method in character recognition is feature extraction. Recognizing the CAPTCHA characters requires recovered characteristics. We can distinguish between the characters thanks to these traits. The structure is built based on the character's physical characteristics. Quantities of horizontal and vertical lines, cross points, endpoints, top and bottom horizontal curves, etc., are a few examples of structural components. The character's shape is then decided to utilize these traits.

Data Splitting

The random state parameter, set to the value 42, divides the data set into the training set and the validation set. The data that will be used to train the neural network and evaluate the model's effectiveness are split into two sets: the training set, which contains 90% of the data, and the validation set, which includes the remaining 10% of the data.

Model Training & Development

The model is constructed using convolutional neural networks to find the CAPTCHA in the image. The model was created using the PyTorch module for the 5-letter captcha pictures. This was accomplished by pre-processing the training sample dataset, after which a model with potentially effective performance layers was built [10]. The batch size is 32, five epochs are utilised, and four layers are in each batch.

Dropout layers, convolutional layers, max-pooling layers, dense layers, and layers with maximum pooling follow the input layer. The number of epochs used for CNN's Model Development of Captcha Images is 150, the batch size is 32, and there are eight layers overall. CNN with FC Classifier uses five epochs, the batch size is 30, and there are 13 layers total. CNN LSTM with CTC operation uses five epochs, the batch size is 100, and there are 11 layers total.

Input Layer: The first layer used in a CNN is the input layer. A Keras tensor is built using the input layer, which also serves to take input in the form of images.

Conv Layer: The kernel or CNN Layer is an alternative term for this layer. CNN's base layer is the layer that takes the input attributes from the image and extracts them. The presence of different convolutional layers is possible; the first layer captures the low-level characteristics of the picture, while the successive layers retrieve the high-level information [11][12].

Fully Connected Layer: The convolutional neural network's last layer is the FC Layer. In this layer, Affine and Non-Linear functions are combined. The fully linked input layer (flatten) takes the result of the preceding layers and "flattens" it. With the help of weights applied to the feature analysis inputs, the first utterly connected layer predicts the proper label. The fully linked output layer provides the final probabilities for each label.

Max-Pooling Layer: The Max Pooling layer delivers its maximum value for the convolutional layer's chunk of the picture it covers. With the help of this layer's dimensionality reduction, all of the input features are compiled into feature maps.

Dense Layer: In our architecture, the neural network's lower half consists entirely of dense layers. A dense layer in a neural network is one whose previous layers are intimately coupled, meaning that each layer's neurons are connected to every other layer's neuron [13].

Flatten Layer: The pooled feature map obtained from the highest pooling layer is flattened in this layer or converted into a lengthy vector suitable for additional processing by the artificial neural network. This makes back propagation easier.

Dropout Layer: This layer, which primarily makes use of regularization strategies, is included to prevent the model from getting excessively fitted. An overfitted model performs poorly in predictions outside of the training set because it can better understand the noise in the data [14].

CTC: CTC is designed to need the text that appears in the image. The dimensions and placement of the characters inside a picture are equally irrelevant. CTC makes the training procedure simpler since it guarantees that an aligned dataset is not required.

LSTM: For storing long-term memories, LSTM networks are excellent. According to the statistics, the network might or might not save memory. The network's Gating mechanisms are responsible for maintaining long-term dependencies. Using a gating mechanism, the network may instantly release or retain memory. Short-term memory impairment plagues recurrent neural networks. They need help to facilitate communication from previous age steps to the latter when a sequence is lengthy enough.

V. Model Evaluation

The best model weights in terms of validation accuracy for predictions during training have been saved using Model Checkpoint, which we utilized to track the performance of our model. It is feasible to evaluate the model's present status while it is being trained at each phase of the process [16]. Using the training dataset, the model may be evaluated to see how well it is "learning."

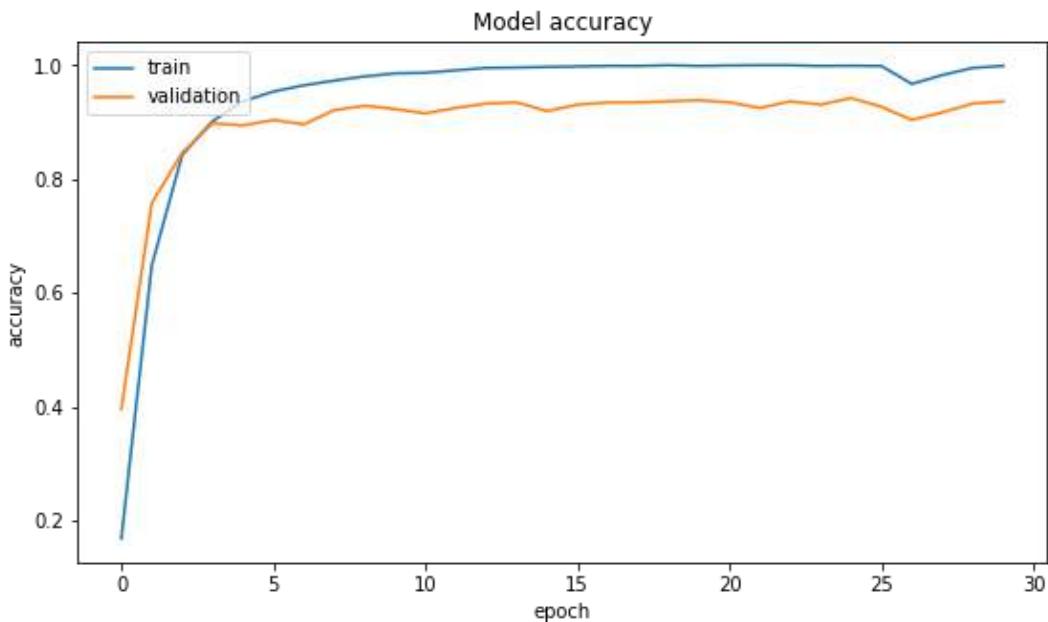


Figure 9 – Accuracy Curve

As can be seen in the images [Figure 9] and [Figure 10], dual learning curves for a CNN model with FC Classifier are typically generated throughout training using both the train and test datasets. A bar graph shown in [Figure 11] depicts the effectiveness of the algorithms used for the research of captcha image recognition and the outcomes of the different methods utilized in this work.

The graph displays each method's accuracy in the order of lowest to highest, as can be seen. CNN and LSTM with CTC operation have a performance of 98 percent accuracy, making it the best model. The bars' height represents each technique's average accuracy. The accuracy rates of the CNN model and the CNN with FC model are 89% and 94%, respectively.

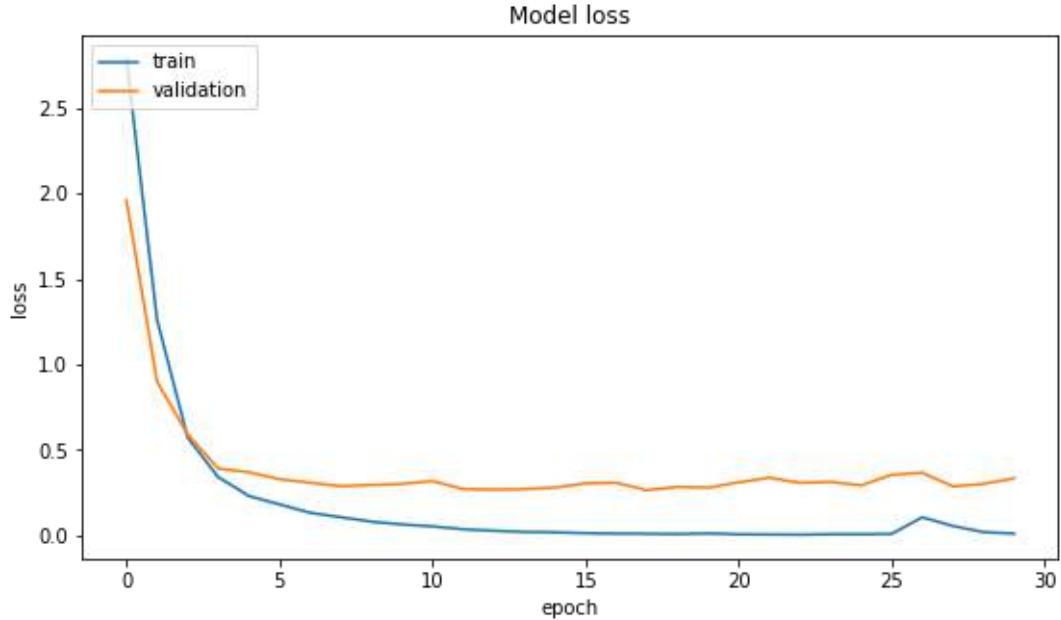


Figure 10 - Loss Curves

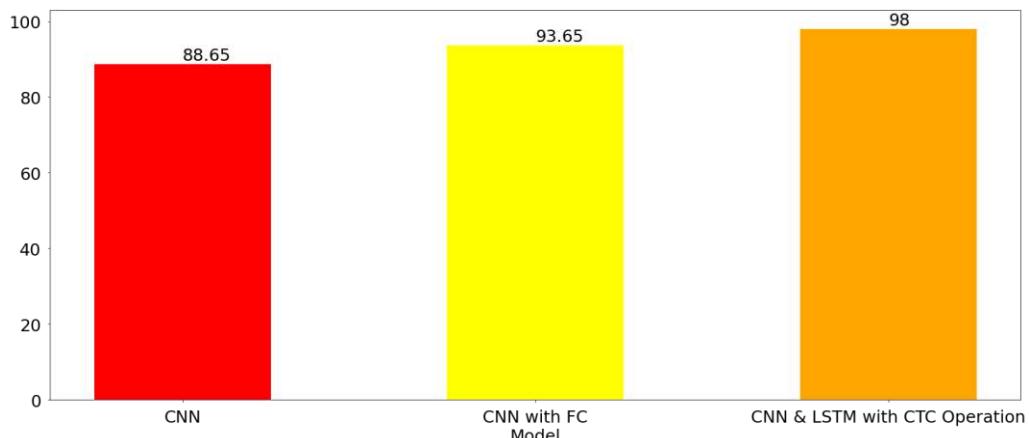


Figure 11 - Accuracies of Classifiers

VI. Conclusion

A test technique called a CAPTCHA is used in network environments to distinguish between people and robots. Studies on CAPTCHA recognition can help identify security vulnerabilities in the CAPTCHA, avoiding hostile network entry. This study suggests a convolutional neural network-based CAPTCHAs identification technology in accordance with the CAPTCHA of pictures character distortion. The experimental study demonstrates that the model developed in this work may improve captcha recognition effects. Although several methods have been put in place to make it more challenging to locate each text in a Captcha picture, and even though the segmentation

approach we utilized can only partially divide characters from one another, our detection method was effective against all targeted schemes. This study proposes a method for identifying CAPTCHAs using convolutional neural networks in line with the CAPTCHA of photographs, and all letters in the images can be recognized. Finally, a few points are raised that need more investigation.

VII. Future Recommendation

Future development will add the Captchas credit of Chinese characters. It is helpful to improve feature extraction precision and deep learning network training methods to avoid incorrectly classifying characters. The model has many parameters, training takes a long time, and there are limits to how much optimization can be done. When using a deep learning network to extract character attributes automatically, characters with similar traits are easily confused. The resilience of other Captcha alternatives and the capacity of new Captcha models to be simultaneously secure and usable are open problems that will be the subject of our future study.

References

- [1] Rich Gossweiler, Maryam Kamvar, and Shumeet Baluja. 2009. What's up CAPTCHA? a CAPTCHA based on image orientation. In Proceedings of the 18th international conference on World wide web (WWW '09). Association for Computing Machinery, New York, NY, USA, 841–850. <https://doi.org/10.1145/1526709.1526822>
- [2] Amit Dhar, Kamalanathan Kandasamy, and Sethuraman Srinivas. 2014. Capturing the Captcha: A Novel Technique to build Self updating Multi-domain Training dataset for Researchers. In Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing (ICONIAAC '14). Association for Computing Machinery, New York, NY, USA, Article 26, 1–7. <https://doi.org/10.1145/2660859.2660935>
- [3] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).
- [4] Rosa Lin, Shih-Yu Huang, Graeme B. Bell, and Yeuan-Kuen Lee. 2011. A new CAPTCHA interface design for mobile devices. In Proceedings of the Twelfth Australasian User Interface Conference - Volume 117 (AUIC '11). Australian Computer Society, Inc., AUS, 3–8.
- [5] Poornachandran, P., Hrudya, P., Reghunadh, J. (2014). Human vs. Machine: Analyzing the Robustness of Advertisement Based Captchas. In: Thampi, S., Gelbukh, A., Mukhopadhyay, J. (eds) Advances in Signal Processing and Intelligent Recognition Systems. Advances in Intelligent Systems and Computing, vol 264. Springer, Cham. https://doi.org/10.1007/978-3-319-04960-1_42
- [6] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.
- [7] Krishna, Ranjay, Ines Chami, Michael Bernstein, and Li Fei-Fei. "Referring relationships." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6867-6876. 2018.

- [8] D. J, S. D. V, A. S A, K. R and L. Parameswaran, "Deep Learning based Detection of potholes in Indian roads using YOLO," 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2020, pp. 381-385, doi: 10.1109/ICICT48043.2020.9112424.
- [9] Y. Wang and M. Lu, "A self-adaptive algorithm to defeat text-based CAPTCHA," 2016 IEEE International Conference on Industrial Technology (ICIT), Taipei, Taiwan, 2016, pp. 720-725, doi: 10.1109/ICIT.2016.7474839.
- [10] P. Kirkbride, M. A. Akber Dewan and F. Lin, "Game-Like Captchas for Intrusion Detection," 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), Calgary, AB, Canada, 2020, pp. 312-315, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00061.
- [11] S. Akshay, T. K. Mytravarun, N. Manohar and M. A. Pranav, "Satellite Image Classification for Detecting Unused Landscape using CNN," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 215-222, doi: 10.1109/ICESC48915.2020.9155859.
- [12] Gao, Haichang, Wei Wang, Jiao Qi, Xuqin Wang, Xiyang Liu, and Jeff Yan. "The robustness of hollow CAPTCHAs." In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, pp. 1075-1086. 2013.
- [13] S. P. Duth and H. L. Rahul, "Identification and Extraction of Text in Non- Blurred and Blurred Image Sequences," 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 615-619, doi: 10.1109/ICICCS48265.2020.9120884.
- [14] Ai, Lazy & Ahn, Luis & Blum, Manuel & Langford, John. (2002). Telling Humans and Computers Apart (Automatically). Communications of the ACM. 47. 10.1145/966389.966390.
- [15] Vinayakumar, R., K. P. Soman, and Prabaharan Poornachandran. "Applying convolutional neural network for network intrusion detection." In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1222-1228. IEEE, 2017.
- [16] Y. Hu, L. Chen and J. Cheng, "A CAPTCHA recognition technology based on deep learning," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 2018, pp. 617-620, doi: 10.1109/ICIEA.2018.8397789.

Basil Leaf Disease Detectionand Classification using Customized Convolutional Neural Network

Deepak Mane^{1*}, Sunil Sangve², Shaila Jadhav³, Disha Patil⁴, Rohan Kakde⁵ and Varad Marudwar⁶

¹²³⁴⁵⁶JSPM's Rajarshi Shahu College of Engineering, Pune, India-411033

¹dtmane@gmail.com

²sunilsangve@gmail.com

³jadhavshaila041@gmail.com

⁴dishakishorpatil@gmail.com

⁵rohankakde1308@gmail.com

⁶marudwarvarad@gmail.com

Abstract.Farming is the mainstream of the Indian economy. Plant leaf disease detection mainly focuses on the health of leaf plants and detecting diseases in their early stages is a tedious task. In this paper, we proposed the Customized Convolutional Neural Network (CCNN) model for predicting diseases of leaves of Basil plants. Proposed model deals. We modified the traditional CNN model in a way that, it deals efficiently with n-dimensional features data as well as unbalanced class data. Here, we predicted diseases of leaves of Basil plants in four categories as Fungal, Downy mildew, Fusarium wilt, and healthy. No such standard dataset is available for Basil leaves so here we prepared our own 916 image data set and the model is trained and tested by different machine learning approaches. However, CCNN has given more accurate results than other existing algorithms. The model detects diseases of leaves with an accuracy of 94.24% for four classes of basil plant leaves.

Keywords: Plant leaf diseases, Image processing, Deep learning, Machine learning, Convolutional neural network.

1 Introduction

Basil (*Ocimum basilicum*) is a popular herb used in many cuisines around the world. It is grown for its fragrant leaves and is an important commercial crop in many countries. However, basil plants are susceptible to a number of diseases that can affect their growth and reduce their yield. Early detection of basil leaf diseases is crucial for controlling and managing these diseases. Traditional methods of disease detection, such as visual inspection and laboratory testing, can be time-consuming and labor-intensive. In recent years, there has been increasing interest in using machine learning techniques, particularly CNNs, to automate the detection of basil leaf diseases. CNNs are a type of DL algorithm that are particularly well-suited for image classification tasks. They have the ability to learn complex features and patterns from images and have achieved state-of-the-art results in a variety of image classification tasks. In the context of basil leaf

disease detection, CNNs can be trained to recognize the characteristic patterns and symptoms of different diseases.

In this introduction, we will discuss the importance of early detection of basil leaf diseases and the potential of using CNNs for this task. In the southeast of Asia, basil is frequently used in cooking as a fresh or dried herb and is also widely utilised in beverages. The leaves can be utilised to extract essential oil, which is then used in fragrances, dental treatments, and cosmetics. Basil also has a number of other benefits, as shown in Fig. 1. Basil plant leaf diseases detection and its classification is complex task. Very few researcher did experiment on medical plant also no such standadrd dataset is available for basil plant. Therefore, we collect and prepared our own Basil plant leaf dataset for detection and classification of leaf diseases. Here, we proposed CCNN model to recognize diseases in four category. Till date no one has tried to detect diseases in four category. Proposed model detect and classify Basil leaf diseases in four category such as Fungal, Downy mildew, Fusarium wilt (yellow leaf)and healthy. Main goal of proposed CCNN algorithm are

- Creating a dataset for Indian Basil plant
- Developing customized CNN for model building.
- Detection and classification of Basil leaf disease in four categories.

Overview of this paper runs as follows: In section 2 it describes related work. In section 3 it describes about system architecture and CCNN model and its working and in section 4 it describes about experimental results and in the last section i.e. section5 we have concluded.



Fig.1. Benefits of Basil

2 Related Work

Numerous researchers have used various machine learning algorithms to identify plant diseases, and work on Indian plant leaf detection and classification is still on-going. In 2012, Piyush Chaudhary et al. proposed a solution to detect diseases on plant leaves. To prepare this model they have used image processing. They have also used colour models like YcbCr, HSI, and CIELB and this model succeeded in identifying the diseases on plant leaves. Image processing and customized machine learning algorithms used to detect diseases on Blueberry plants [2]. The technique used by the author is Deep learning in which they used Convolutional Neural Network (CNN). To enhance and modify an image they used different image filtering techniques. They were able to perceive whether a blueberry plant leaf is affected by any disease or not. The accuracy of the system is 84%. S. Sladojevic et al. proposed a system which recognises plant diseases by classification of leaves images in Deep Neural Networks. They develop models which recognize 13 different types of plant disease. This Experiment shows the accuracy from 91% to 98% [3]. In paper [4], Developed a deep CNN model in which five different diseases of tomato leaves were identified after conducting extensive research. The model proposed by Belal A. M et al. Infected tomato leaves were clearly distinguished from healthy tomato leaves using the newly created model, which had 99.84 percent accuracy. For sugar beet disease detection and differentiation, the author used Support Vector machine algorithms in [5]. According to different types and stages of disease the average accuracy of the model is between 65% to 90%. There are various complicated problems in exploration of different areas like in the biology sector, biomedicines. Plant Disease Classification and Detection by using convolutional neural network (CNN) which is a type of deep learning algorithm that utilizes machine learning (ML). (CNN) model is proposed by Muhammad H. Saleem, Johan Potgieter and Khalid Mahmood in [6]. They use a plant village dataset and experimental accuracy of the model is 90%. In [7], the authors used CNN architecture on a plant village dataset of sugarcane. YOLOv3 and Faster-RCNN algorithms are used for getting more accuracy. The result of the sugarcane leaf disease detection model is 93.20%. In [8], M. Nagaraju used deep learning convolutional neural network, Artificial intelligence, DCNN, (ANN) techniques to design a system which detects plant leaf disease. They used Plant village, Real field Dataset and average accuracy of model is 91.53%. In [9], Authors used three methods: 1. Single Image Prediction 2. Graphical interface (GUI) Method 3. Through Local Host disease detection. Plant leaf is detected with overall accuracy of 95%. In [10], Rinu R, Manjula S H has proposed a system which detects the plant leaf disease with accuracy of 94.8%. System also works in unfavourable conditions using convolutional neural network. Real life leaf images were captured and used to train the model based on CNN by authors in [11]. To divide the leaf into unhealthy and healthy category, additional pre-processing techniques are used and the accuracy achieved by Rishiikeshwer, B.S. et al. is 95% on actual leaf images and 98% on standard dataset. In [12], S. SanthanaHariet al. proposed a model which detects the plant disease of multiple plants, like Apple, Maize, Tomato, Grape,

Potato in early stages. Technique used by the authors is CNN and accuracy achieved is 86.00%. In [13], Sammy V. Militante et al. has proposed a model which has achieved 96.5% rate of accuracy, using Deep Learning, Computer Vision, CNN on plant village dataset. Tomato plant diseases are identified by authors in [14], using the image processing as the techniques and CNN algorithm. The accuracy achieved is 98%. In [15], Deep learning-based disease recognition models for plants include a variety of qualities, including the fact that they are unsupervised, have high accuracy, have strong universality, and have great training efficiency. The accuracy of the model is on average 83.75 percent. In [16], In order to classify and identify basil leaves disease, the survival of fittest approach was used along with 3 class data i.e. 2 diseased classes that are Downy mildew, leave spot and 1 healthy class they have got accuracy of 95%.

3 System Architecture

The entire accomplishment procedure is broken down into the essential stages in the subsections below.

3.1 Data-set preparation:

At every stage of illness identification research, from the training phase to assessing the effectiveness of classification algorithms, an appropriate dataset is very necessary. It should be noted that there has been very limited work done on Basil Plant Leaf and whatever work that has been done, they have not offered access to their data-set, so due to the lack of a data-set to train the proposed model. As a result of this study, a self-made basil leaf image dataset has been created and used. A sample collection of 916 basil leaf samples has been taken from a variety of locations including farms and houses. There are four classes in the data-set: Fungal, Downy mildew, Fusarium, Wilt (yellow leaf), and Healthy leaves. The RGB image is then converted into a gray-scaled images by converting it from RGB to gray-scale. At first, each image was cropped separately. Each of these cropped samples is then re-sized to a 256 x 256 size image. The images are renamed and saved separately for each of them. In order to increase the size of the data-set, various transformations have been performed via scaling: every image is. Due to these transformations, the data-set size has increased by four times. Some sample images from our dataset are represented in Table 1 and its types of diseases with symptoms represented in Table 2. Dataset is classified into four classes as three diseased and one healthy. In order to classify leaf images, we use their symptoms as criteria for classifying them. During the creation of the Basil dataset, images were divided into different groups based on the symptoms they displayed. Images are categorized as Fungal, Downy mildew, Fusarium wilt (yellow leaf) and healthy leaves in four classes. Every class contains a different number of images which is as follows:

- Fungal: 253
- Downy mildew: 175
- Fusarium wilt (yellow leaf): 212
- Healthy: 276

Table 1: Data-Set Sample

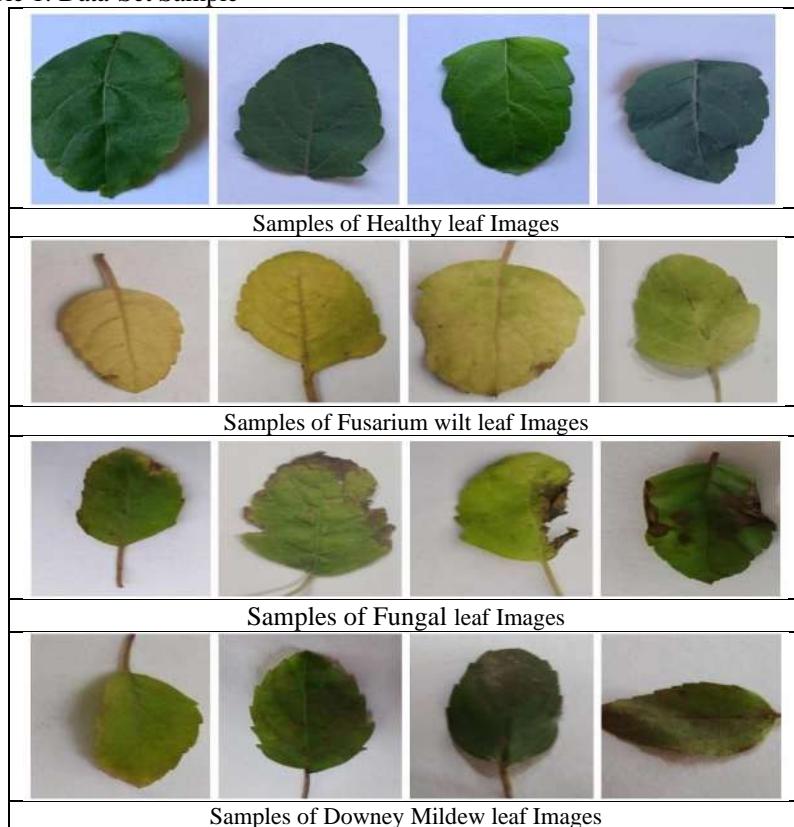


Table 2– Types of disease and symptoms

Leaf Image				
Diseases & its Symptoms	Fungal : The dark spots on leaves can be circular or of any shape with a light center or a dark center. It causes a fungus on leaves.	Downy mildew : Plants may get dark to black angular necrotic areas, grey fuzzy on the underside of the leaves, and discoloration around the main vein that spreads outward.	Fusarium wilt : A plant with dark streaks on the lower side of the leaf, yellow, withering leaf, and stunted growth.	Healthy Leaf : Healthy leaves are green in color and do not exhibit any dark spots.

3.2 Experimental process of the model

Three layers are included in the proposed model, namely the convolution layer, the pooling layer, and the fully connected layer. As in fig. 2, there are two more key parameters in addition to these three layers, namely the dropout layer and the activation function. Block diagram for the working is shown in Fig. 2. When it comes to solving classification problems, CNN is probably one of the most powerful and accurate method [17][18]. We achieved the high accuracy for model by following some tuning parameters like: By collecting more data, More layers can be added by adding more filters, by Increasing or Decreasing the Size of the Image and in order to increase or decrease the number of epochs.

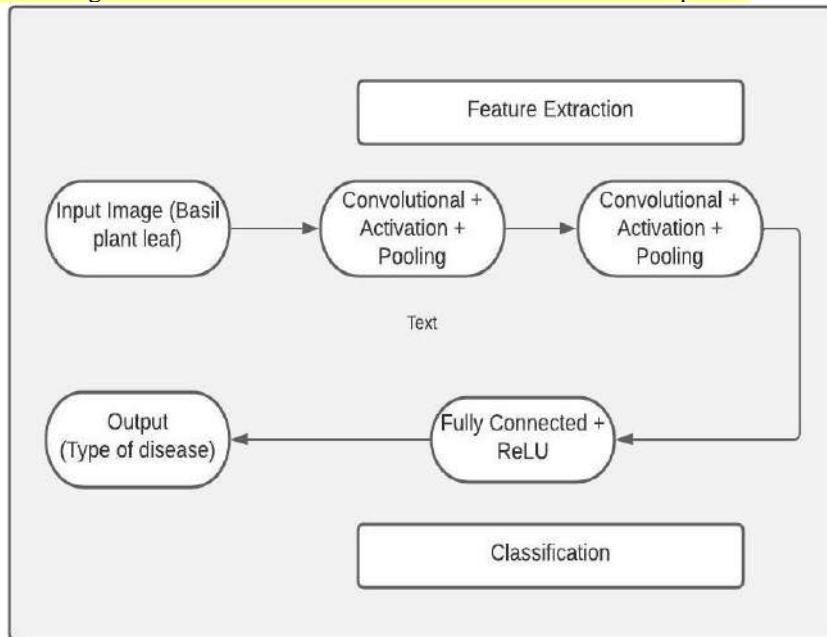


Fig 2: Work Flow Block Diagram

3.4 System Architecture

Due to the large number of parameters in a leaf image, CNN's are very effective for image classification. Using CNNs to reduce parameter numbers without compromising model quality is an extremely effective method. In order to reduce the dimensions of the input matrix, a sliding window of smaller size is used. Creating models with CNN is a straightforward process, especially when it comes to n-dimensional data and big data. Quite a few researchers have experimented with basil as a medicinal plant, but there is no standard dataset for it. To detect and classify leaf diseases, here we collected and prepared our own basil leaf dataset. In this paper, we proposed a CCNN model to classify diseases into four categories. Until now, no one has attempted to detect diseases in four categories.

Proposed CCNN model is predicting diseases of leaves of Basil plants. Proposed model efficiently deals with n dimensional features data as well as unbalance class data. Firstly we are providing the Basil plant leaf dataset to the system and input image of disease leaf. In feature extraction, it will draw out characteristics from an image that can be used to interpret it. Then, during feature selection, a subset of relevant features (variables, predictors) will be chosen for use in building the model. We will explore deep learning techniques to automatically classify and identify plant diseases using leaf photos. Comparison of diseases will identify the presence of leaves and distinguish between healthy leaves and various diseases. Then a specific plant disease will appear. Architecture diagram for the proposed model is shown in Fig. 3.

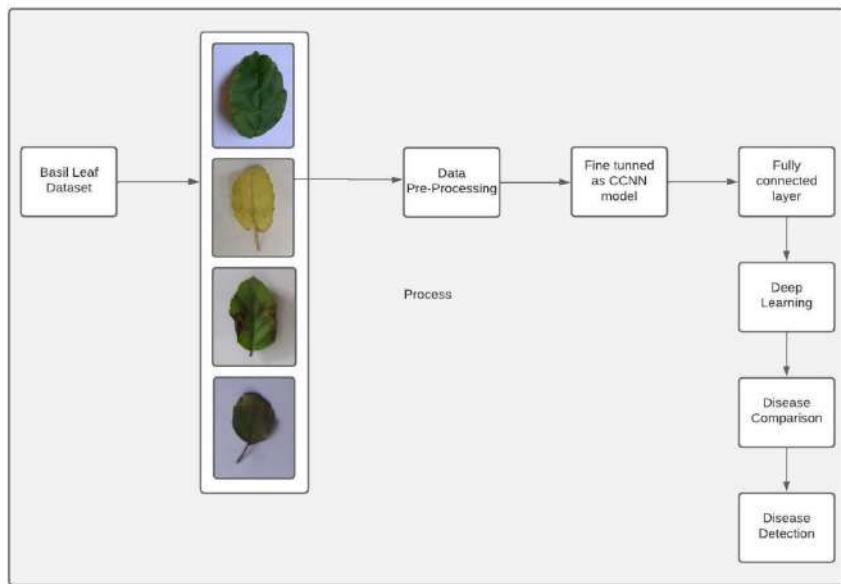


Fig 3: Architecture Diagram

3.5 Methodology

The following steps are used to detect leaf diseases using an input image.

3.5.1 Image acquisition

Image acquisition (Data preparation) is retrieving an image from some source. Any vision system starts with the acquisition of images. The process of taking an image with a digital camera and saving it in a digital format is known as image loading. Using a camera, we typically take images of both healthy and unhealthy. Images are captured from the agricultural field as we have build our own dataset. The images are of RGB (Red, Green, and Blue) format.

3.5.2 Image pre-processing

The image data is improved through image pre-processing, which reduces unwanted distortions. We perform image processing in order to remove unwanted distortion and noise which enhances the image by using techniques like image enhancement, RGB to Lab conversion, filtering etc. Pre- processing includes image clipping, smoothing, and enhancement. Denoising can be performed using a variety of reduction techniques. With the right threshold, medium filters perform better with salt and pepper noise.

3.5.3 Image segmentation

Image segmentation involves dividing an image into sections with similar pixel compositions. The segmentation of the image into several segments will depend on the area of interest. Image segmentation is the process of dividing an image into similar areas. There are many ways to segment data, including Otsu's approach, k-means clustering, and transforming RGB images into HIS models. For each set of characteristics, the k-means cluster algorithmic rule is used to categorise objects into K different categories. By reducing the sum of the squares of the distance between the object and the matching cluster, objects are classified.

3.5.4 Feature extraction

Feature extraction extracts features that can be used to determine the meaning of a given image. After the image segmentation, the disease portion of the image is extracted. Color, texture, shape, edges, and morphology are characteristics that can be used in plant disease detection. For feature extraction, the Gray-Level Co-occurrence Matrix (GLCM), which depicts spatial arrangements and separation between features, is used. The statistical features that were extracted are mean, variance, skewness, kurtosis, eccentricity, contrast, energy, perimeter, homogeneity, area, standard deviation, centroid, aspect ratio, entropy, and sum of entropy.

3.5.5 Classification & Detection of Diseases

Classification is correctly predicting the value of a discrete class variable or attributes. Since most green pixels refer to the healthy leaf and are of no use for disease identification procedures, this method, which is used to shorten processing time by deleting the green pixels from the leaf, is used. Computing the intensity value of the green pixels allows for the masking of those pixels. The RGB component of a given pixel is given a value of zero if the intensity is below a predetermined threshold value.

3.6 Forward steps for CCNN learning

Here, CCNN model involves five layers, that is: convolutional layer and pooling layer, flatten layer, dropout layer and dense layer.

3.6.1 Convolutional layer

The first layer used to extract the different features from the input images convolutional layer. Layers of convolutional neural networks are responsible for storing the output of the prior layer's kernels, which consist of weights and biases for learning.

As a fundamental component of CNN architecture, convolution layers are used. A combination of linear and nonlinear operations is used to extract features in this method. The activation function is used in conjunction with convolution to extract features. Feature extraction from leaves is carried out by a convolution layer in the CNN.

$$x_{ij}^l = \sum_{m=0}^{a-1} \sum_{n=0}^{a-1} \omega_{mn} y_{(i+m)(j+n)}^{l-1}$$

.....(equ. 3.6.1)

3.6.2 Pooling layer

An initial layer of convolution is followed by an additional layer of pooling. The purpose of this layer is to reduce overfitting and also reduce the feature map size and training-time

Rectified linear unit (ReLU) is used as an activation function in the pooling layer. In ReLU, negative pixels are set to zero element-by-element. A rectified feature map is generated as a result of introducing non-linearity to the network. In comparison with sigmoid and tanh functions, ReLU is considered to be six times faster in its convergence. The ReLU function reduces the amount of time it takes to calculate because no exponential terms are incorporated into the formula.

$$\text{ReLU function: } R(z) = \max(0, z)$$

.....(equ. 3.6.2.1)

A stride of k can be used to transform an input image A ($n_1 \times n_2$) into an output image by applying either the max function fmax or the average function favg to each $k \times k$ block in the image. The max function fmax returns the maximum value in each block, while the average function favg returns the average value of the pixels in each block. This process reduces the size of the image while retaining important information.

3.6.3 Dropout layer

As a result of connecting all features to the fully connected layer, the training dataset tends to overfit. An overfitted model performs very well on training data but it fails on testing data which is a negative impact on the model. For overcoming this problem we have to use dropout layer which drops some neurons during training which reduces size of the model. For instance, when the neural network passes a dropout of 0.2, 20% of the nodes are randomly removed. A dropout layer enhances a machine learning model's performance by preventing overfitting by simplifying the network.

3.6.4 Flatten layer

The Flatten layer is used to converts the $28 \times 28 \times 50$ output of the convolutional layer into a single one-dimensional vector which is used for input to the dense layer. The most parameters are in the final dense layer. Each and every output ‘pixel’ from the convolutional layer is connected to the 10 output classes via this layer.

3.6.5 Dense layer

In dense layer each neuron receives input from every neuron in the previous layer, giving the layer its name as dense. Based on the results of convolutional layers, pictures are categorized using dense layers.

4. Experimental Results

In [16], the authors propose a model for the detection of Basil leaf disease, using their self-made dataset of Basil plant leaves. The model they have developed is capable of detecting both healthy leaves and two different basil diseases, including Downy mildew and Leaf spot. Their trained model has an overall accuracy of 95.73 percent. We have created our own basil dataset with 916 leaf images since the basil plant leaf dataset is not available as an open source. In the proposed method, the average accuracy achieved is 96.66% with 3 basil leaf classes and 94.24% with 4 basil leaf classes. There are three types of healthy leaves and two types of unhealthy diseases, Downy mildew and Fusarium wilt. The four classes are divided into 1 healthy class and 3 different leaf diseases, Downy mildew, Fusarium wilt, and Fungal wilt. The proposed model can detect 3 different

varieties of basil leaf disease, as well as normal leaf. This makes it more effective and flexible enough to be used for a wide range of diseases. Table 3 depicts comparison between existing set of model and our proposed model.

A comparison of the proposed CCNN basil plant leaf disease detection model with contemporary methodologies [16]

Classification models [16]	Test Set Accuracy
Random Forest	94.31%
Naives Bayes	91.86%
KNN	92.43%
Support Vector Machine	92.89%
Discriminant Analysis	92.30%
Bayesian Generalized Linear	72.04%
Gaussian Process	89.57%
Extreme Gradient Boosting	93.87%
Conditional Inference Tree	93.36%
Flexible Discriminant Analysis	92.42%
Proposed Model-CCNN	94.24%

Table 3. Analyses of comparison between the proposed model and the existing classification model.

Results on different models:

During the development of this model, multiple transfer learning techniques and models were used to detect basil plant leaf disease. We obtained different kinds of results with varying levels of accuracy. Accordingly, we conclude that the customized CNN model provides the highest accuracy of all. Fig. 4. Compares the results of transfer learning models with those of CCNN model.



Fig 4 Results comparison with different transfer learning approaches

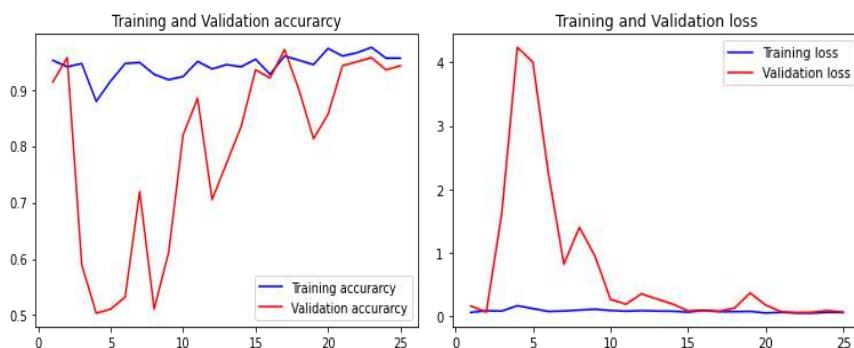


Fig 5. performance graph between Training vs Validation

For four (Fungal, Downy mildew, Fusarium, healthy) diseases got 94.24% accuracy and on three (Downy mildew, Fusarium wilt, healthy) got 96.6 % accuracy. Performance between Training verses validation loss graph represented in Fig. 5 and classification report represented in Fig. 6. From the confusion matrix it is clearly evident that the dataset and the model which we have built are giving great results as we can see for Downy Mildew disease 32 samples of 35 were predicted correctly, similarly for fungal 21 samples out of 25 were predicted correctly where as for fusarium wilt 16 of 16 were predicted correctly and for healthy class 62 out of 63 were predicted correctly. Confusion matrix is represented in Fig. 7.

	precision	recall	f1-score	support
0	1.00	0.91	0.96	35
1	0.88	0.84	0.86	25
2	0.80	1.00	0.89	16
3	0.98	0.98	0.98	63
micro avg	0.94	0.94	0.94	139
macro avg	0.91	0.93	0.92	139
weighted avg	0.95	0.94	0.94	139
samples avg	0.94	0.94	0.94	139

Fig. 6 Classification report

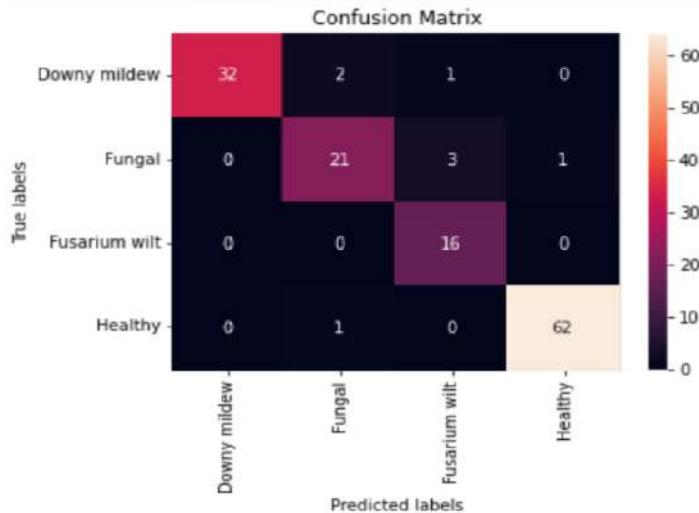


Fig. 7 Confusion matrix

5 Conclusion

This paper includes implementation procedure, materials, techniques of plant leaf disease detection model on Basil leaf using CNN technique and for implementing the proposed model we have created our own dataset using images of basil plants. Convolutional Neural Networks (CNNs) have shown to be effective in the detection of plant leaf diseases as we have achieved accuracy of 94.4% for Basil plant using our own dataset. They have the ability to learn complex features and patterns from images, which makes them well-suited for Basil plant disease detection. In recent years, numerous studies have demonstrated the effectiveness of CNNs in detecting various plant diseases, including diseases of fruits,

vegetables, and grains. These studies have used a variety of CNN architectures, including VGG, ResNet, and Inception, and have achieved high accuracy rates in their respective datasets. One of the main advantages of using CNNs for plant leaf disease detection is their ability to handle large amounts of data and to automatically learn features from the images. This can significantly reduce the need for manual feature engineering and make the process more efficient. Overall, CNNs have proven to be a powerful tool for Basil plantdisease detection and have the potential to greatly improve the efficiency and accuracy of disease detection in agriculture.

References

1. Piyush Chaudhary, Anand K. Chaudhari, Dr. A. N .Cheeran, and Sharda Godara, “Colour transform based approach for disease spotdetection on plant leaf,” *International Journal of Computer ScienceandTelecommunications*, vol. 3,no.6, pp.65–69,2012.
2. Sumair Aziz, et. al. “Image Pattern Classification for Plant Disease Identification using Local Tri-directional Features” ,*IEEE10thAnnual Information Technology, Electronic and Mobile Communication Conference (IEMCON),2019*.
3. Srdjan Sladojevic et. al. "Deep neural networksbased recognition of plant diseases by leaf image classification,"*Computational Intelligence and Neuroscience*, vol.11,2016.
4. Belal A. M. Ashqar, Samy S. Abu-Naser (2018), “Image Based Tomato Leaves Diseases Detection Using Deep Learning”, *International Journal of Academic Engineering Research (IJAER)*,Vol. 2No.12,pp.10-16.
5. T. Rumpf , Anne-KatrinMahlein,U. Steiner,E.-C. Oerke,H.-W.Dehne,Lutz Plümer “Early detection and classification of plantdiseaseswithSupportVectorMachinesbasedonhyperspectralreflectance”,2010.
6. Muhammad Hamma Saleem, Johan Potgieter and Khalid Mahmoodrif, “PlantDisease Detection and Classification by Deep Learning”,*plants*,vol8,468,Oct-2019.
7. Malik HashmatShadab et al. “Disease Recognition in SugarcaneCrop Using Deep Learning” *Advances in Artificial Intelligence andData Engineering*,pp189–206,2020.
8. M.Nagaraju, Priyanka Chwla“ Systematic review of deep learning techniques inplant disease detection”*IntJSystAssurEngManag*,vol11(3),pp547–560,May2020.
9. Mohit Bansal et. al. “Plant Leaf Disease Detection and Recognition”. *International Journal of All Research Education and Scientific Methods*, Volume 9, Issue 6, June -2021
10. Rinu R, Manjula S H,“ Plant Disease Detection and Classification using CNN”, (IJRTE) ISSN: 2277-3878 (Online), Volume-10 Issue-3, September 2021.
11. Rishiikeshwer, B.S., T. AswinShriram, J. Sanjay Raju, M. Hari, B. Santhi and G.R. Brindha , SASTRA Deemed University, Thanjavur India,” Farmer-Friendly Mobile Application for Automated Leaf Disease Detection of Real-Time Augmented Data Set using Convolution Neural Networks”pala.
12. S.SanthanaHari et. al. “Detection of plant disease by leafimage using convolutional neural network”, 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN).
13. Sammy V. Militante1, Bobby D. Gerardo, Nanette V. Dionisio“Plant Leaf Detection and Disease Recognition using Deep Learning” published in 2019 IEEE Eurasia Conference on IOT, Communication and Engineering, ISBN: 978-1-7281-2501-5.
14. Surampalli Ashok et. al. “Tomato Leaf Disease Detection Using Deep Learning

- Techniques”, IEEE Conference Record # 48766; IEEE Xplore ISBN: 978-1-7281-5371-1.
- 15. Yan Guo et. al “Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming”, Hindawi Discrete Dynamics in Nature and Society Volume 2020.
 - 16. Dhingra, G., Kumar, V., & Joshi, H. D. (2019, March 15). Basil leaves disease classification and identification by incorporating survival of fittest approach. ELSEVIER.
 - 17. Mane, D. T. and U. V. Kulkarni. "A Survey on Supervised Convolutional Neural Network and Its Major Applications." *Deep Learning and Neural Networks: Concepts, Methodologies, Tools, and Applications*, edited by Information Resources Management Association, IGI Global, 2020, pp. 1058-1071. <https://doi.org/10.4018/978-1-7998-0414-7.ch059>
 - 18. Mane, Deepak et. al (2022). Detection of Anomaly using Machine Learning: A Comprehensive Survey. *International Journal of Emerging Technology and Advanced Engineering*. Vol. 12, Issue 11pp-134-152. 10.46338/ijetae1122_15.

Segmentation of Brain Tumours from MRI Images Using CNN

*Ilango, D¹ and Sulthana, R¹

¹ Department of Computer Science, Birla Institute of Technology & Science, Pilani - Dubai Campus, Dubai, United Arab Emirates

^{1*}dhakshinailango@gmail.com

² Department of Computing and Mathematical Sciences, University of Greenwich, Old Naval Royal College, London-SE10 9LS, United Kingdom

²razia.sulthana@greenwich.ac.uk

Abstract. Early detection of brain tumours is key to proper diagnosis and treatment. It can be classified as malignant or benign based on the aggressiveness of the tumour. To diagnose a patient, an MRI imaging device is used to obtain scans of the brain. Due to the large quantity of data produced, radiologists must perform the tedious task of going through each MRI image to identify the brain tumour's location, size, and origin. This process is prone to human error and is also time-consuming. Therefore, this paper proposes a methodology to accurately diagnose and segment the brain tumours from the MRI images using Convolutional Neural Networks (CNN) specifically U-NET architecture.

Keywords: Machine learning, Biomedical Image Segmentation, Brain Tumours, Convolutional Neural Networks.

1 Introduction

Brain tumours occur when an abnormal division of these cells which leads to a cluster of cells inside the brain. Some of the main symptoms of brain tumours are fatigue, difficulty in coordination, speech difficulties, confusion, seizures, headaches, nausea or vomiting, issues in vision etc. They can be classified as primary and metastatic brain tumours. Primary is when cancer originates from the brain and does not spread to any other parts of the body. And Metastatic brain tumours are when cancer originates from some other part of the body to the brain. Tumours can also be benign which means non-cancerous or malignant which means the brain tumour is cancerous in nature [1].

There are many kinds of Brain Tumours depending on location, origin, size etc. Some of the common types of Brain Tumours are:

1. Glioma – it is a common type of brain tumour that originates from the brain. The tumour arises from the aggressive division of glial cells that surround neurons. These can be malignant or benign depending on the spread rate. About 33 % of brain tumours are gliomas.

2. Meningioma – This is also a very common brain tumour that originates from the brain and 30% of brain tumours and meningiomas. It originates in the meninges which consist of the layer of tissues that surround and cover the brain just under the skull. Most Meningiomas are benign but some can be persistent and return even after treatment [2].
3. Gliosarcoma – These tumours are connected with other supportive tissues. They are mostly benign but can spread to other areas. They are also usually aggressive and resistant to chemotherapy [2].

To detect and diagnose a patient with brain tumours usually, MRI imaging devices are utilized as they provide better results with respect to soft brain tissue as compared to other types of imaging devices. The MRI device makes use of the hydrogen molecules present in our body and as per the principles of electromagnetic imaging, any moving charge has its own magnetic field. Therefore, an MRI device with the help of powerful magnets and external radiofrequency and sensors connected to a computer can obtain images of the brain. To analyse these images radiologists, must sift through numerous MRI sequences to accurately diagnose the size, type, and origin of the brain tumour in the patient. The various MRI sequences are T1, T1C, T2 and Flair. Each sequence provides additional information about the tumour [3].

To avoid manual MRI image segmentation of brain tumours the proposed methodology uses a Deep Learning model to segment and classify the data. Neural Networks are a set of supervised learning algorithms that mimics the human brain in the way it operates. The nodes simulate the neurons in the brain and the connections between them act like synapses in the brain. It consists of three main layers input, hidden and output layers. Neural Network becomes a Deep Learning model when it has several hidden layers in it. Specifically, Convolutional Neural Networks (CNN) are achieving great results in the field of computer vision and biomedical image processing. An image is taken as input in the form of a matrix of pixels. The model makes use of filters for example of size 3x3 and slides them across images for automatic extraction of features. For each pixel, a value is computed using a convolutional operation. This process drastically reduces the number of inputs and weights required [4]. The output of the convolution operation is passed to an activation function which determines the presence of the essential feature required. In a similar manner, the number of layers implemented can be increased to check the presence of nuanced features [5].

This paper aims to solve the tedious task of manual segmentation of brain tumours from MRI images. This process can be automated by using CNN. In the Literature survey, this model had the greatest accuracy among all the other segmentation methods. It was also found that using a less complex CNN can decrease the computational overhead of training a Deep Learning (DL) model as well as increase the accuracy compared to other Machine Learning algorithms. The issue of

diagnosing brain tumours with variable sizes, shapes and structures can also be achieved [5].

2 Related Work

The first study [6] proposed an approach for the segmentation of brain tumours using a thresholding algorithm which classifies voxels above a specific threshold as containing tumours. The proposed method can be condensed into three stages. The first stage is pre-processing which improves the quality of the MRI image by using techniques such as morphological and pixel subtraction operations. The second stage is the segmentation stage which is a threshold-based segmentation. Its main function is to transform grayscale images into binary images. The third stage is the image filtering stage which uses a median filter to eliminate noise from the images. Finally, this approach has a success rate of 94.28 % and an overall accuracy of 96% [6].

This study [7] provided an approach that enhanced the DL algorithm which is a kernel-based CNN with M-SVM. In the pre-processing stage, the LoG and CLAHE methods are used to detect the edges of images. They are also used to remove unwanted noise and parts of the background from the MRI images. Feature extraction is done using the SGLDM method. It is based on the spatial distribution of levels of Gray in the regions of interest which focused on the construct, site, and overall shape of the tumour in the image. The M-SVM algorithm is used to classify the image as either ordinary or unusual. The images in the unusual class are passed as input to the kernel-based CNN which isolates the tumour in the image. Finally, this segmented image is classified as either malignant or benign [7].

This paper proposed [8] a two-step verification method for brain tumour image segmentation wherein 2 algorithms are used. The Watershed algorithm and SIFT together are called as Watershed-matching algorithm. Brain tumour image segmentation is done using the classical watershed algorithm. SIFT is used to match the segmented brain tumour image to the original image. This is the first verification step. Next, the volume of the tumour is calculated to distinguish between benign and malignant tumours. The area of the tumour is calculated and on detection of white pixels greater than 500, the tumour is classified as being in the critical stage else the initial stage. Hence the paper uses a two-step verification process for brain tumour image segmentation [8].

This paper [9] detailed an improved approach to brain tumour image segmentation using the Fuzzy C Mean algorithm and multiobjective optimization. The Fuzzy

Clustering algorithm is used to cluster a set of data points so that it has more similarity with one group than the other. Segmentation of brain tumours from MRI images is done using Fuzzy C Means and Genetic Algorithm. To extract objects from the background Thresholding segmentation is done. Finally, classification is performed by using an ensemble of Decision trees [9].

In this paper [10] a novel Dominant Gray Level Based K-Means Algorithm is used as opposed to a standard K-Means Algorithm which randomly chooses k number of pixels as the centroid. The proposed methodology first uses pre-processing techniques to convert red, green, and blue images of the MRI Brain image dataset to grayscale images. The algorithm is utilized to discover the probabilities of individual pixels. In this K-Means clustering algorithm, the top 16 pixels with the highest probabilities are chosen as centroids initially [10].

In [11] the aim was to perform automatic brain tumour segmentation in MRI images using less computational costs and a texture and contour-based algorithm. The first stage is the detection of tumour slices. This is done by creating a histogram for each hemisphere of the brain and determining which hemisphere would likely contain tumours. Feature extraction is done using AM-CWT and DT-CWT and other statistical techniques. Feature selection is done using a modified Regularized Winnow Algorithm. Finally, the tumour region is segmented by using an active contour model Skippy Greedy Snake Algorithm [11].

In [12] a Multi-Atlas Based Adaptive Active Contour Model was used for brain tumour image segmentation in MRI and CT images. The method utilizes theoretical information for the segmentation of brain tumours. It consists of reference images and label maps. The STAPLE algorithm is used to combine the label maps into a more accurate brain tumour contour. Active contour models are classified as either edge or region-based models. The proposed AELR active contour model is implemented using the DRLSE algorithm [12].

In [13] multi-modality MRI imaging scans were used to perform brain tumour image segmentation. Firstly, feature extraction is performed on pre-processed images. All the features combined are given as input to the RF Classifier to predict five classes namely background, necrosis, oedema, enhancing tumour and non-enhancing tumour. Finally, the class labels are used to hierarchically calculate three regions - complete tumour, active tumour and enhancing tumour [13].

This study [14] presented a comparison of various dimensionality reduction PCA Algorithms such as PPCA, EM-PCA, GHA and APEX with two clustering algorithms K-Means and Fuzzy C-Means Algorithm. PCA is essential to decrease

the complexity of the image dataset as well as combat the curse of dimensionality issue. It was observed that PPCA achieved the best results out of all the clustering methods and the combination of EM-PCA and the PPCA with the K-Means algorithm obtained the best results for clustering and segmentation [14].

This research paper [15] focussed on the identification, segmentation, and early detection of brain tumours by using feature extraction methods such as DWT, GLCM and Probabilistic Neural Networks as a classifier. The first stage is the pre-processing of images which included Segmentation and Region growth. The feature extraction is performed by using DWT for extracting coefficients of wavelets and GLCM for extraction of statistical features. Finally, the images are given as input to the PNN for the classification of abnormal and normal brain tissues. The location of the tumours are also identified [15].

This study [16] aimed to implement Deep Learning Neural Networks to distinguish between ordinary brain tissues and different types of tumour tissues using brain MRI scans. In the proposed methodology image segmentation is performed using the Fuzzy C-Means algorithm. Feature extraction is done using DWT and dimensionality is reduced using PCA. Finally, the classification of the brain tumours is done using Deep Neural Networks (DNN) [16].

This [17] research paper focused on the fusion of various MRI sequences to provide an accurate analysis of the brain tumour. They are combined using DWT and Daubechies wavelet kernel into a single MRI image. Next PDDF is performed for noise reduction and global thresholding. Lastly, the images are classified using a CNN which consisted of 23 layers. These layers include - convolutional, batch normalization, Rectified Linear Unit function, max pooling for downsampling, fully connected and SoftMax layers to detect images with the presence of tumours [17].

The central idea of this study [18] was to improve the existing CNN model by using the BAT algorithm which is used for the automatic segmentation of brain tumours from images. An optimized loss function is used to achieve this. The two phases of this model are pre-processing and segmentation. A hybrid CNN model is used and the Loss function is optimized by making use of the BAT algorithm [18].

In [19] prior knowledge is introduced in the CNN model. The information given was that most of the tumour regions in the images are of left-right symmetry. This prior knowledge is ignored by most of the existing CNN models. Therefore, by using DCSNN brain tumour image segmentation is performed by using multiple symmetric masks in the layers. The model is computationally efficient as it took only 10.8 s to segment an image. There are many stages in the proposed method.

The first stage is data pre-processing. Next is the baseline network which integrates a bottom-up and top-down pathway. The DCSNN takes four modal MRI images, so the Left-Right Similarity Mask (LRSM) presents a broad asymmetrical situation of four MRI sequences. The DCSNN is a five-classification task model. Finally, post-processing is done to remove segments with sparse voxels [19].

Research paper [20] aimed to present a CNN model that is less complex and has greater picture clarity and accuracy. The backpropagation method is used to classify brain tumours. The objective is to check if the model can classify test cases effectively into regular or irregular classes. To achieve this the final layer was executed only in the training phase. The convolutional layers consider the local spatial constructs in the previous layers. The forward pass is when multiple layers of neurons are interlinked in a way that each successive layer just relays the information from the previous layer which is what takes place in a feed-forward network. The Backward pass is when the weight or bias is modified, and the shift is propagated backwards all throughout the network for changing the weights and biases. The last layer consists of a soft-max function [20]. Table 1. shows a summary of all of the various algorithms that can be used for brain tumour image segmentation.

Table 1. Different methods used for brain tumour image segmentation

References	Problem Statement	Dataset	Algorithm	Advantage	Performance Measure Value
[6]	To achieve clear segmentations of brain tumours that can be used by a medical professional to gain more insight about the tumours to aid with their diagnosis.	TCIA (The Cancer Imaging Archive, 2017)	Thresholding Algorithm	Easy and efficient implementation	96%
[7]	Difficulty in brain tumour segmentation in an MRI image due to non-uniform boundaries of tumours and location of tumours in the brain.	Not mentioned	Laplacian of Gaussian filtering method (LoG) with Contrast Limited Adaptive Histogram Equalization (CLAHE),	Effective segmentation increased accuracy and low time complexity.	84 %

			Convolutional Neural Networks (CNN) with Multiclass-Support Vector Machine (M-SVM)		
[8]	Issues with diagnosing brain tumours present in the soft tissues of the brain as biopsy of tumour tissues are required which is time-consuming as well as prone to errors.	BRATS 2012 dataset	Watershed Algorithm and SIFT (Scale-Invariant Feature Transform) Algorithm	Computationally less complex and non-invasive.	98.5 % accuracy
[9]	Challenges regarding accurate segmentation of MRI images of brain tumours due to spatial and structural differences. Hard to isolate a region of interest.	DICOM MRI database	Fuzzy C-Means algorithm, Genetic Algorithm and Decision Trees	Improvement in the centre of cluster detection and increased convergence time.	92 %
[10]	Addresses the issue of selecting a random point as centroids for clustering which results in imprecise brain tumour image segmentation.	MRI Brain image Database	Dominant gray level-based K-Means Clustering Algorithm	Able to characterize regions with increased accuracy and less computational complexity.	95.37 %
[11]	The huge computational cost issue with implementation of existing techniques for the automatic detection of brain tumours is addressed.	NCI-MICCAI, 2013 database	Regularized Winnow, k-NN Classifier and Skippy Greedy Snake Algorithm	Increased Segmentation Accuracy, and reduced computational costs.	93.82 %
[12]	The difficulties with respect to exact delineation and the introduction of interobserver and intraobserver variability of brain tumours in radiotherapy.	Shandong Cancer Hospital and Institute (Shandong, China) in 2019	Simultaneous truth and performance level estimation (STAPLE) and the distance regularized level	Time-saving and accurate segmentation.	87.19 %

			set evolution (DRLSE) Algorithm		
[13]	Usage of multi-modality images to classify brain tumours.	MICCAI BraTS 2013	Random Forest Classifier Algorithm	Multi-modality MRI images are used to give an accurate account of the location and size of the tumour and increase classification accuracy.	88% - complete tumour region, 75% - core tumour region and 95% - enhancing tumour region
[14]	In image segmentation increase in the number of features selected leads to a curse of dimensionality problem and affects the overall performance of the algorithm.	Dataset of T1w MRI images were obtained from a patient who suffered from a brain tumour	Conventional Principal Component Analysis (PCA), Probabilistic Principal Component Analysis (PPCA), Expectation Maximization Based Principal Component Analysis (EM-PCA), Generalize Hebbian Algorithm (GHA) and Adaptive Principal Component Extraction (APEX) algorithms, K-Means and Fuzzy C-Means Algorithms.	Comparison of various PCA algorithms in combination with FCM and K-Means Algorithm, less time consuming and less complex image data.	Error rates for 512x512 images for PCA -3.7993, EM-PCA - 3.7430, PPCA - 3.7991 , GHA - 4.7339 AND APEX - 4.5778
[15]	Difficulty in identification, segmentation, and discovery of affected areas in brain tumour MRI images at an early stage.	Digital Imaging and Communication s in Medicine (DICOM) dataset	Discrete Wavelet Transform (DWT), Gray-Level Co-Occurrence Matrix (GLCM)	Fast and high accuracy in the detection of brain tumours at early stages.	95% accuracy

and Probabilistic Neural Networks (PNN)					
[16]	Issues differentiating between normal brain tissues and other specific brain tumour tissues.	Dataset obtained from Harvard Medical School Website	Fuzzy C-Means (FCM), Discrete Wavelet Transform (DWT) and Deep Learning Neural Networks Algorithms	Differentiate between different types of brain tissues and able to identify complex relationships in the dataset.	96.97 %
[17]	Single MRI sequences do not provide enough information for accurate diagnosis of the type, shape, and severity of brain tumour	BRATS 2012, BRATS 2013, BRATS 2015, BRATS 2013 Leader board and BRATS 2018	Discrete Wavelet Transform (DWT), Daubechies wavelet kernel, PDDF, Global Thresholding Algorithm and Convolutional Neural Networks (CNN).	More information is provided due to the fusion of 4 MRI sequences, High Accuracy, and Large data set.	0.97 ACC - on BRATS 2012 Image, 0.98 ACC - BRATS 2013 Challenge, 0.96 ACC - BRATS 2013 Leader board, 1.00 ACC - BRATS 2015 Challenge and 0.97 ACC - BRATS 2018 Challenge datasets.
[18]	The structure of tissues adjacent to the tumours is changed by the tumour mass effect. Therefore, an Improved CNN model is required which produces optimized MRI images.	Brain Tumour Segmentation Challenge 2015 database (BRATS 2015)	Enhanced Convolutional Neural Networks (ECNN) and BAT Algorithms	Increased accuracy compared to that of a regular CNN model.	92 %
[19]	The changing nature of brain tumours with respect to size, shape and location affects the accuracy of segmentation	BRATS 2015	Deep Convolutional Symmetric Neural Network (DCSNN)	Complex function mapping achieved which learns features by itself, less computation time.	0.852 dice similarity index
[20]	A Less complex CNN model with high accuracy and speed	Radiopaedia 2013 and BRATS 2015	Convolutional Neural Network (CNN)	Training data are not corrupted by outliers, are less	97.5 % accuracy

complex and
have high
accuracy

3 Proposed Work

The Brain tumour Image Segmentation is done using T1w MRI scans and using the RSNA MICCAI PNG Dataset from Kaggle which consists of training and testing data [21]. The Algorithm used is Convolutional Neural Networks specifically U-NET architecture. This is done by using Transfer Learning. The U-NET model has been trained on the BraTs 2020 Dataset [22]-[24].

The U-NET model is specifically used for biomedical image segmentation. The advantage of using this algorithm is that it can be trained end to end using very few images compared to other CNN architectures. It is U-shaped and symmetric and works by using an encoder – a decoder path or is also known as a contracting and expansive path. The encoder half works by reducing the spatial dimensions of the image as it passes through each layer [25].

The Encoder path is responsible for feature extraction. With each convolution operation, detailed features are extracted and the input size of the image is reduced. In the Encoder path, the convolutional operation takes place by convolving the input MRI scans using an $n \times n$ size filter to give an output feature map, Batch Normalization normalizes the output of the convolution layer that is it brings the mean to 0 and standard deviation to 1. Finally, the output from the Batch Layer is passed onto the Rectified Linear Unit Function (ReLU) Activation Function.

The Max Pooling Layer present in the Encoder Path is used to calculate the maximum value from each 2x2 size window of the feature map. The Convolution method which consists of the convolution layer is called multiple times with a different number of filters starting from 32, 64, 128 and 256.

The Decoder Path is responsible for the reconstruction of the segmented masks from the extracted features. In the Decoder path, the spatial dimensions of the image are increased, and the value of the number of channels is reduced. This also starts with the calling on the Convolution method with a different number of filters but in this case, the number of channels gradually decreases as opposed to Max Pooling where it gradually increased in number. Upsampling is done each time after the convolution method is called. It is done to preserve the input volume size at the end of each convolutional operation. The feature maps of the encoder are concatenated

with the decoder so that the model can segment the brain tumours from the MRI scans using detailed and general features. Finally, the input is passed to the final convolutional layer with kernel size 1x1 and with the Sigmoid function as an Activation function.

In this project, the U-NET model is implemented by using the concept of Transfer Learning which is using prior knowledge of the model when solving one problem and using that information to solve a similar application. Fig 1. Gives an overview of the proposed methodology.

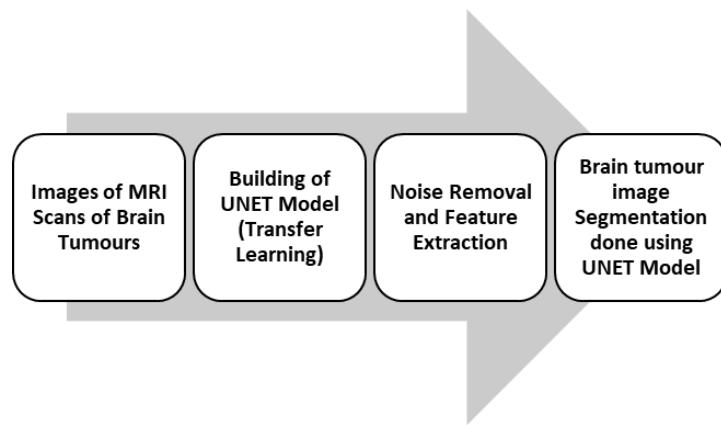


Fig. 1. Architecture Diagram

Brain Tumour Image Segmentation:

The prediction of the tumour is done by first resizing the image to 240 x 240 and then the model is used to predict the mask using the resized image and the dimension of the image is reversed by using the transpose method. The noise from the image is removed by using morphological functions in using OpenCV library. Opening which is Erosion followed by Dilation was performed to remove unnecessary minute objects from the image and to smoothen the boundary of the tumour. This function to predict the tumour is used in the find_tumour function which finds the path where the image on which the prediction should be made and passes it to the prediction function. It also returns the path to the tumour slice if the tumour pixel is greater than max_detected. Then a display function is used to display the image of the brain in grayscale and then the tumour in red.

4 Results Analysis

The model was able to accurately predict brain tumours as the accuracy of the U-NET model is 98%. The presence of a tumour in the MRI Scan was predicted by using a red mask as shown in Fig 2. The model was able to precisely segment tumours of varying boundaries and edges.

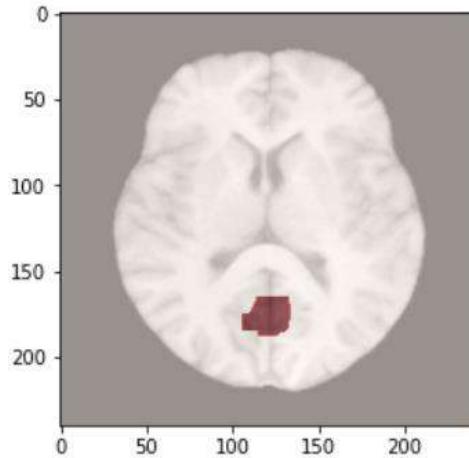


Fig. 2. Brain Tumour Image Segmentation using MRI Scans

5 Conclusion

In Conclusion, we find that Deep Learning architectures, specifically U-NET architecture is accurate in biomedical image segmentation. Using this method variable size, shape and structure tumours can be easily diagnosed. And there will be no requirement for manual segmentation by radiologists as well. In the future, the model can be further optimized by trying different loss functions and optimizers as well as testing the model on different modalities of MRI Scans [26]-[28].

References

- [1] Mayo Clinic Staff, Brain Tumour, Retrieved October 15th,2021, from <https://www.mayoclinic.org/diseases-conditions/brain-tumor/symptoms-causes/syc-20350084>
- [2] The John Hopkins University, Types of Brain Tumours, Retrieved Octob er 15th, 2021, from <https://www.hopkinsmedicine.org/health/conditions-and-diseases/brain-tumor/brain-tumor-types#malignant>
- [3] National Institute of Biomedical Imaging and Biomedical Engineering, Magnetic Resonance Imaging (MRI), Retrieved October 15th 2021, form <https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri>
- [4] Stewart, M., Simple Introduction to Convolutional Neural Networks, Retrieved October 20th 2021, from <https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac>
- [5] Lang, R., Zhao, L., & Jia, K. (2016, October). Brain tumor image segmentation based on convolution neural network. In 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering, and Informatics (CISP-BMEI) (pp. 1402-1406). IEEE.
- [6] Ilhan, U., & Ilhan, A. (2017). Brain tumor segmentation based on a new threshold approach. Procedia computer science, 120, 580-587.
- [7] Thillaikkarasi, R., & Saravanan, S. (2019). An enhancement of deep learning algorithm for brain tumor segmentation using kernel based CNN with M-SVM. Journal of medical systems, 43(4), 1-7.
- [8] Hasan, S. M., & Ahmad, M. (2018). Two-step verification of brain tumor segmentation using watershed-matching algorithm. Brain informatics, 5(2), 1-11.
- [9] Rodríguez-Méndez, I. A., Ureña, R., & Herrera-Viedma, E. (2019). Fuzzy clustering approach for brain tumor tissue segmentation in magnetic resonance images. Soft Computing, 23(20), 10105-10117.
- [10] Nitta, G. R., Sravani, T., Nitta, S., & Muthu, B. (2020). Dominant gray level based K-means algorithm for MRI images. Health and Technology, 10(1), 281-287.
- [11] Nabizadeh, N., & Kubat, M. (2017). Automatic tumor segmentation in single-spectral MRI using a texture-based and contour-based algorithm. Expert systems with applications, 77, 1-10.
- [12] Zhang, Y., Duan, J., Sa, Y., & Guo, Y. (2020). Multi-Atlas Based Adaptive Active Contour Model with Application to Organs at Risk Segmentation in Brain MR Images. IRBM.

- [13] Usman, K., & Rajpoot, K. (2017). Brain tumor classification from multi-modality MRI using wavelets and machine learning. *Pattern Analysis and Applications*, 20(3), 871-881.
- [14] Kaya, I. E., Pehlivanlı, A. Ç., Sekizkardeş, E. G., & Ibrikci, T. (2017). PCA based clustering for brain tumor segmentation of T1w MRI images. *Computer methods and programs in biomedicine*, 140, 19-28.
- [15] Shree, N. V., & Kumar, T. N. R. (2018). Identification and classification of brain tumor MRI images with feature extraction using DWT and probabilistic neural network. *Brain informatics*, 5(1), 23-30.
- [16] Mohsen, H., El-Dahshan, E. S. A., El-Horbaty, E. S. M., & Salem, A. B. M. (2018). Classification using deep learning neural networks for brain tumors. *Future Computing and Informatics Journal*, 3(1), 68-71.
- [17] Amin, J., Sharif, M., Gul, N., Yasmin, M., & Shad, S. A. (2020). Brain tumor classification based on DWT fusion of MRI sequences using convolutional neural network. *Pattern Recognition Letters*, 129, 115-122.
- [18] Thaha, M. M., Kumar, K. P. M., Murugan, B. S., Dhanasekeran, S., Vijayakarthick, P., & Selvi, A. S. (2019). Brain tumor segmentation using convolutional neural networks in MRI images. *Journal of medical systems*, 43(9), 1-10.
- [19] Chen, H., Qin, Z., Ding, Y., Tian, L., & Qin, Z. (2020). Brain tumor segmentation with deep convolutional symmetric neural network. *Neurocomputing*, 392, 305-313.
- [20] Suneetha, B., Rani, A. J., Padmaja, M., Madhavi, G., & Prasuna, K. (2021). Brain tumour image classification using improved convolution neural networks. *Applied Nanoscience*, 1-9.
- [21] U. Baid et al., “The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification,” arXiv:2107.02314 [cs], Sep. 2021, [Online]. Available: <https://arxiv.org/abs/2107.02314>
- [22] B. H. Menze et al., “The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS),” *IEEE Transactions on Medical Imaging*, vol. 34, no. 10, pp. 1993–2024, Oct. 2015, doi: 10.1109/tmi.2014.2377694.
- [23] S. Bakas et al., “Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features,” *Scientific Data*, vol. 4, no. 1, Sep. 2017, doi: 10.1038/sdata.2017.117.
- [24] S. Bakas et al., “Identifying the Best Machine Learning Algorithms for Brain Tumor Segmentation, Progression Assessment, and Overall Survival Prediction in the BRATS Challenge,” arXiv:1811.02629 [cs, stat], Apr. 2019, [Online]. Available: <https://arxiv.org/abs/1811.02629>

- [25] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” arXiv.org, May 18, 2015.
<https://arxiv.org/abs/1505.04597>
- [26] A. R. Sulthana and A. K. Jaithunbi, “Varying combination of feature extraction and modified support vector machines based prediction of myocardial infarction,” Evolving Systems, Jan. 2022, doi: 10.1007/s12530-021-09410-4
- [27] P. R, S. P, R. S. A, and V. T, “Brain Tumor Segmentation and Prediction using Fuzzy Neighborhood Learning Approach for 3D MRI Images,” Jul. 2021, doi: 10.21203/rs.3.rs-497725/v1.
- [28] R. Pitchai, P. Supraja, A. R. Sulthana, T. Veeramakali, and Ch. Madhu. Babu, “MRI image analysis for cerebrum tumor detection and feature extraction using 2D U-ConvNet and SVM classification,” Personal and Ubiquitous Computing, Mar. 2022, doi: 10.1007/s00779-022-01676-y.

Distributed Training of Large Scale Deep Learning Models in Commodity Hardware

*Jubaer Ahmad, Tahsin Elahi Navin, Fahim Al Awsaf, Md. Yasir Arafat,
Md. Shahadat Hossain, **Md. Motaharul Islam

Dept of Computer Science and Engineering, United International University,
Dhaka, Bangladesh

({jahmad181023, tnavin191056, fawsaf181253, marafat183030,
mhossain173032}@bscse(uiu.ac.bd)

**Corresponding author: motaharul@cse.uiu.ac.bd

Abstract. Running deep learning models on a computer is often resource intensive and time-consuming. Deep learning models require high-performance GPUs to train on big data. It might take days and months to train models with large datasets, even with the help of high-performance GPUs. This paper provides an affordable solution for executing models within a reasonable time interval. We propose a system which is perfect to distribute large-scale deep learning models in commodity hardware. Our model consists of creating distributed computing clusters using only open source software which can provide comparable performance to High Performance Computing clusters even with the absence of GPUs. Hadoop clusters are created by connecting servers with a SSH network to interconnect computers and enable continuous data transfer between them. We then set up Apache Spark on our Hadoop cluster. Then we run BigDL on top of Spark. It is a high-performance spark library that helps us scale to massive datasets. BigDL helps us run large deep learning models locally in Jupyter notebook and simplifies cluster computing and resource management. This environment provides computation performance up to 70% faster than a single machine execution with the option of scaling in case of model training, data throughput, hyperparameter search and resource utilization.

Keywords: Hadoop, Apache Spark, BigDL, SSH.

1 Introduction

The amount of data produced each day is only increasing. Around 41 billions of devices will be interconnected, producing 79 zettabytes of data [1]. To handle such a big amount of data, high performance GPUs will play a vital role. GPU prices are on the rise and the shortage of GPUs continues to increase. Under such circumstances, an affordable alternative for data processing can play a vital role in

the management of big data [2]. The concept of sharing processing and storage power among multiple devices to execute huge datasets are getting popular each day. Over the past few years quite a few projects such as in [3] [4] on cluster computing have been launched.

We chose Hadoop cluster for our computer clusters because of its capacity to manage large amounts of data. Hadoop clusters are affordable, and they can simply be scaled up by adding more cluster nodes without requiring changes to the application logic. Hadoop clusters are also resilient due to their replication feature. With commodity hardware, any corporation may build up a robust Hadoop cluster. The basic role of a Hadoop cluster is to store data on worker nodes.

The master node will store metadata and manage the cluster system as a whole. Apache Spark is a fast and general processing framework compatible with Hadoop clusters. We must execute it on top of the Hadoop cluster to process data in HDFS. We will run spark on standalone mode where the master and each worker have their own web UI that displays cluster and job information. Spark offers machine learning and deep learning with advanced analytics features. Because of its low-latency in-memory data processing capacity [5], it can tackle a wide range of analytics challenges. BigDL [6] is a distributed deep learning package that runs on Apache Spark, and will be used for our research. It uses existing Spark clusters to execute deep learning computations and makes data loading from large Hadoop datasets easier.

The following points summarize the major contributions of our paper:

- We have developed an automated system that allows users to effortlessly divide deep learning workloads over numerous computers for enhanced performance in deep learning. The cluster communication will execute via LAN connectivity.
- This system allows consumers to obtain a performance boost with the addition of extra logical cores.
- This automation system enables customers to run large-scale deep learning models on available and affordable commodity hardware.
- Our system can speed up deep learning research by 20% for users who do not have access to expensive hardware such as GPU.
- We have developed an automatic bash script generator to develop highly complex scripts.

The rest of the paper is organized as follows: in Section II, we address the literature review, in Section III, we suggest a methodology, in Section IV, we design the implementation system, in Section V, we include simulations and evaluations, and in Section VI, we conclude the paper.

2 LITERATURE REVIEW

For the execution of large data analysis or deep learning models, distributed computing is an emerging technology. This technology enables large data processing and management in the workplace without the use of physical high-performance computers. By using this technology some low-performance computers can be connected together and become a powerful computing machine.

LadonSpark, A free and open-source method for automatically configuring and deploying Spark cluster. In this system Hadoop distributed file systems can be deployed automatically [1]. DeepSpark [2] a distributed and parallel deep learning system that uses both Apache Spark and Caffe. They used Apache Spark to create the deep learning framework interface and parameter exchanger. Hadoop and Spark are assessed and compared in this paper in terms of runtime, memory and network utilization and central processor efficiency [3]. SystemML is used for expressing machine learning algorithms in a higher-level language and compiling and running them in a mapReduce environment[4].

BigDL is a framework, they [5] introduce it including its distributed execution approach, compute performance, training scalability, and real-world use cases. MLLib [6], Spark's distributed machine learning library with implementations of common learning methods such as classification, regression, collaborative filtering, clustering, and dimensionality reduction. In [7] introduces MPCA SGD, a method for distributed deep neural network training on top of the Apache Spark. DeepSpark [8], a DL framework built on Spark and developed an asynchronous stochastic gradient descent (SGD) for better DNN training.

They [9] show how to improve ML on Apache Spark by employing MapRDD and the newly developed asynchronous stochastic gradient descent (SGD) method. In [10], they deployed a framework for DNN training using Spark which can train large scale deep networks using computer clusters with thousands of devices. They present a unique multimodal [11] DNN based on the CNN architecture for remote sensing observation classification. In [12], they compare all Deep Learning frameworks, conduct experiments, and assess the outcomes with a comprehensive table.

3 PROPOSED METHODOLOGY

Using BigDL with Apache Spark and Hadoop, training of large scale deep learning models can be distributed among the computers in a cluster. First we create a user interface where we have the option to write host IP, hostname to configure hostname and host file of each PC. Then we designate Master and workers. Then Install OpenSSH server automatically on each PC and Generate SSH key for each PC pairs and check SSH passwordless connectivity. After the

connection is established, User will install python and Install Jupyter Notebooks by himself on Master Node. Then we configure hadoop files, Spark and BigDL and the script will be generated automatically.

According to Figure 1, first, we establish a cluster of machines over a secure shell connection in our proposed approach. Using encryption and hashing techniques, a secure shell connection ensures data integrity and privacy. It connects computers and allows them to interact using hypertext, similar to web pages. It authenticates users by issuing each node with a unique SSH key. Following the completion of the connection between PCs, we use HDFS to store our data in a distributed manner. We create a multi-node Hadoop cluster that includes a master node and two worker nodes. Apache Spark is used for data analysis and processing. The master node monitors and controls the worker nodes as they process data in parallel. Spark jobs created by the context are distributed to worker nodes by the driver. Each worker node's executor executes tasks based on job instructions. Spark incorporates outputs created by the executor in each node via spark driver in addition to data processing. The BigDL library is then installed on Apache Spark. We can use the BigDL library to execute deep learning models..

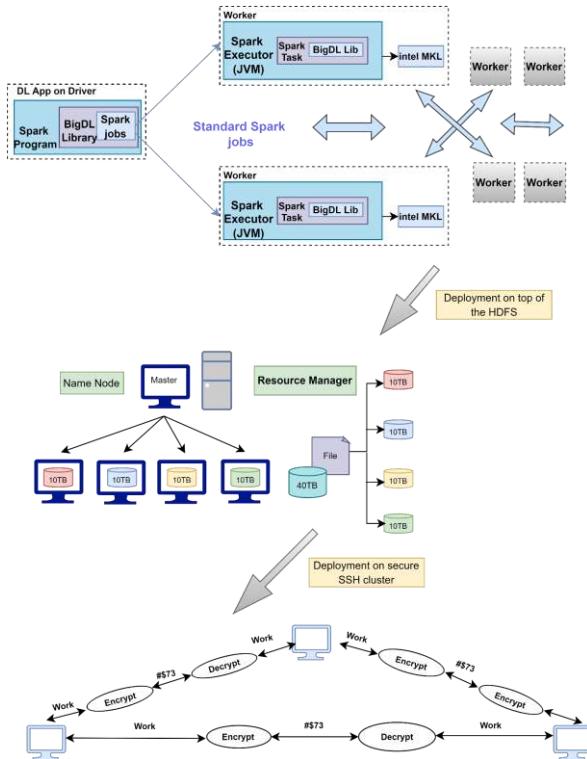


Figure 1: Implementation Architecture

3.1 Graphical User Interface using Automation Scripts

We create a GUI for setup automation of all the softwares and packages. This makes the setup easy for users who are not familiar with configuration using XML and other text based configuration files. We use Zenity to create GUI based bash automation scripts [20]. A single shell script is run to complete the whole setup without directly having to edit any configuration files. The shell script starts running in the backend. Using Zenity, users are prompted to provide network details of each computer in the cluster such as device hostname, device IP address, and other parameters such as number of master and worker devices and each of their unique configuration details such as datanodes, masternodes, secondary masternodes, resource managers etc.

After collecting details from the GUI, they are stored in a SQLite database for backup. These details are then used as variables to be placed into other configuration files which are automatically accessed through the original shell script.

3.2 Secure Cluster Management

Data transfer between network devices must be managed in order for devices to communicate appropriately. A set of rules known as a network protocol regulates the communication. Telnet and RSH (remote shell) were early network protocols that did not provide enough security against malicious cybersecurity threats. The SSH protocol was prompted by the demand for a more secure network communication technique. SSH, also known as Secure Shell or Secure Socket Shell, is a network protocol that allows us to safely access and communicate with remote computers (mostly remote servers) [16]. SSH creates a cryptographically secure connection between the client and server, authenticating each side and transferring commands and output back and forth. We create RSA key based passwordless connectivity between each device in the cluster to ensure secure and uninterrupted connectivity between them. This enables only the devices which have authorized each other's keys to access the network without password.

3.3 Distributed file management

Our proposed method requires Hadoop's HDFS as a distributed data store and Apache Spark to process data by distributing the tasks among each node by creating JVM tasks.

We are storing data in a Hadoop cluster because Hadoop's HDFS is a great option to store large amounts of data in distributed storage setups. It provides

especially high throughput when the file sizes are large and the amount of data is high [19]. Hadoop clusters are easily scalable and can quickly add nodes with increasing data blocks. Hadoop clusters can be set up in commodity hardware which is an inexpensive solution for a distributed file system. It also replicates the data across multiple nodes which reduces the chance of cluster failure.

As depicted in Figure 2, to create a Hadoop cluster, first a collection of machines is needed so that we can create a system of interconnected nodes that helps run an application by working together.

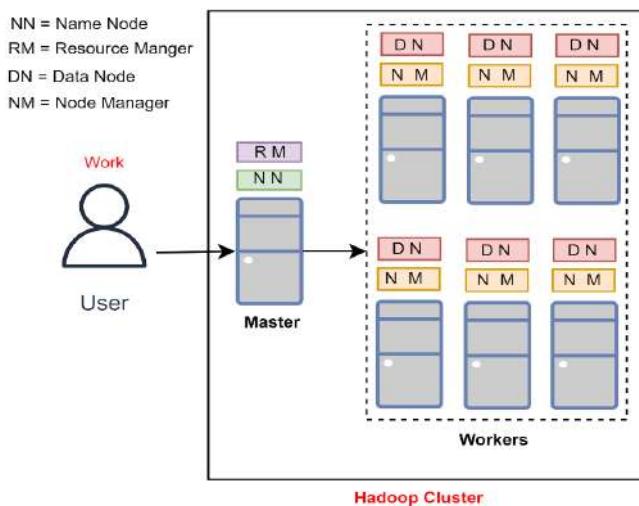


Figure 2: HDFS file storage using Hadoop cluster

3.4 Spark Process Management

We can use Hadoop map-reduce to run deep learning models directly on a cluster using frameworks such as DeepLearning4J, but we'll use Apache Spark on top of Hadoop because it outperforms the Hadoop map-reduce system and it is much more user friendly [15]. We'll go through why Spark is utilized and how it works to run massive data processing tasks and deep learning models.

Spark is a big data processing framework that utilizes “In-Memory” cluster computing. In-memory computing refers to the use of middleware software that enables data to be stored in RAM across a cluster of machines and processed in parallel.

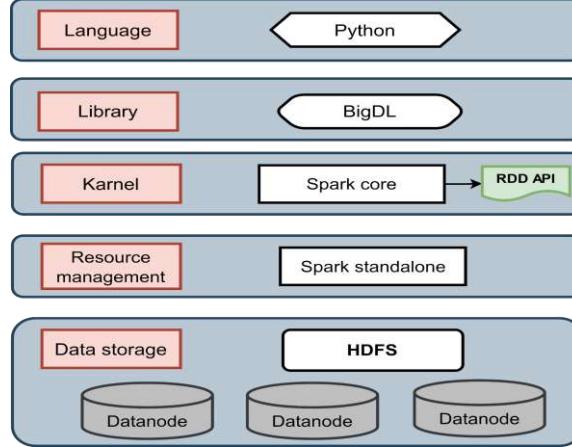


Figure 3: Spark on top of hadoop

Figure 3 shows the relation between Hadoop and Spark in a cluster setup. First, we set up HDFS to store data in the cluster. Then we use Spark as a standalone cluster manager. The Spark Core serves as the backbone of the entire Spark project. It includes a variety of functionality, such as task management, scheduling, and input-output procedures. BigDL and Spark MLlib help Spark Core for training Deep Learning. BigDL solves the deep learning dependency problem. Different parts of Figure 3 are described below:

Spark Standalone and Spark Core: It denotes that Spark is built on top of HDFS and that HDFS is given special consideration. In this case, Spark and MapReduce will run in parallel to cover all Spark jobs on the cluster.

Data Sharing using Spark RDD: Spark's concurrent and fault-tolerant data format is the Resilient Distributed Dataset (RDD). Spark sends RDD data to the cluster automatically and executes concurrent actions on it.

BigDL: Users can create any deep learning applications as ordinary Spark programs using BigDL, which is constructed as a library on top of Spark [17].

In the following Figure 4, we have the control panel master, spark driver, and python IDE which is inside the master node. Using the control panel master we manage the data distribution and execution among local machines and the spark driver communicates with the datanodes to keep track of the information about the Executors at all times. The cluster manager creates a number of Spark workers. The spark executor is a part of the spark core residing in each datanode that executes local machine data individually.

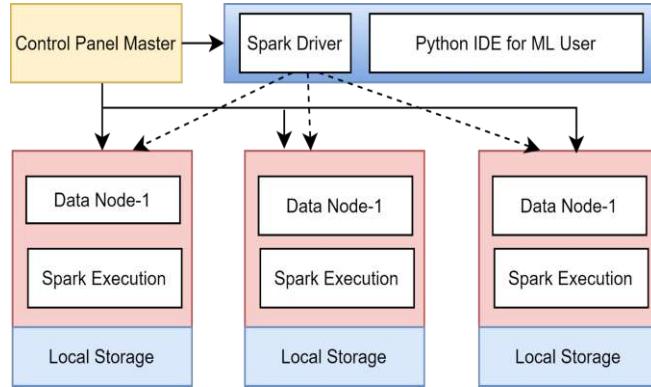


Figure 4: Spark Execution Model

3.5 Distributed deep learning management

BigDL is a distributed deep learning library developed by Intel and provided to the open source community with the goal of bringing large data processing and deep learning together. BigDL's purpose is to make deep learning more accessible to the Big Data community by allowing them to construct deep learning applications using common tools and infrastructure.

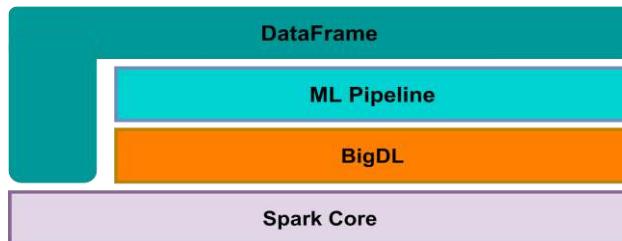


Figure 5: BigDL in relation to Apache Spark components

BigDL [18] is implemented as a library on top of Spark, as seen in Figure 5, allowing users to construct deep learning applications as ordinary Spark programs. As a result, it can work in tandem with other Spark libraries (such as Spark SQL and Dataframes, Spark ML pipelines, [14] Spark Streaming, Structured Streaming, and so on) and run on existing Spark or Hadoop clusters.

A variety of machine learning algorithms in Spark MLlib are based on synchronous mini-batch SGD (Stochastic Gradient Descent). These algorithms leverage Spark's reduce or tree Aggregate methods to aggregate parameters.

4 SIMULATIONS AND EVALUATIONS

I.

We divide this section into two parts. First we describe our experimentation environment. Then we describe our experimentation results.

4.1 Simulation setup

We have based our experiments on three types of environment setups, which are most commonly used and accessible by students. The competition of our setup is Google Colab (Free Tier), which is most commonly used by students and also anyone who doesn't have access to expensive computational resources. The other competition is a simple local machine setup, comparable to a single laptop or a desktop machine. Our model is implemented on a 3 node cluster, which is one of the best ways to train large scale deep learning models in our opinion, because distributed computing can be easily implemented by students in their computer labs or even connecting their personal computers together so that they can make better use of the resources already available to them. Increasing data size. Table I shows a comprehensive configuration of our 3 test environments.

Device Type	Processor	GPU	RAM	Physical Core
3 Node Cluster	Intel i7 - 8 th Gen	None	8X3 = 24GB	8X3 = 24 @ 2.4 GHz
Google Colab	Intel Xeon	NVIDIA P100 - 16GB	13 GB	1@2.4GHz
Single Server	Intel i7-8th gen	None	8GB	8@2.4GHz

Table I: Experimentation Environment Configurations

4.2 Evaluation Result

As we see in the following comparisons between the 3 setup environments previously mentioned, the 3 Node Cluster using Apache Spark and Intel BigDL. Our experimentation focuses on two common metrics: Hyperparameter tuning and Model Training Performance.

Hyper-parameter Tuning Performance: Hyper-parameter tuning is a common step for improving machine learning pipelines. The traditional method of hyperparameters optimization is a grid search, which simply makes a complete search over a given subset of the hyperparameters space of the training algorithm. In this work, we have used Spark to speed up this step. Spark allows training multiple models with different parameters concurrently on a cluster, with the result of speeding up the hyperparameter tuning step. Performance of hyperparameter tuning with grid search runs multiple training jobs in parallel,

therefore it is expected to scale well when run in parallel, as it was the case in our work (see also the paragraph on hyperparameter tuning). This is confirmed by measurements, as shown in Figure 6, by adding executors, i.e. the number of models trained in parallel, the time required to scan all the parameters decreases, as expected. For a single number of execution elapsed time is more than 8 minutes and it decreases with the increment of the executor and it reduces time with 4, 2, 0.9 minutes for node 2, 3 and 4 respectively.

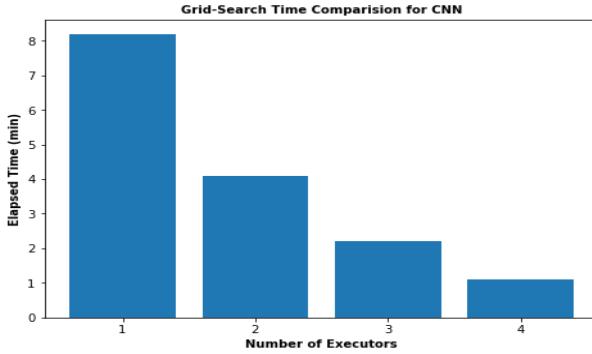


Figure 6: Hyperparameter Optimization Performance using GridSearch

CPU utilization: The MNIST classifier is appropriate for training on desktop computers due to the training dataset size being small in scale. However, the model complexity is high for the B-RNN model. Figure 7 depicts the CPU usage while training the MNIST classifier on a Spark cluster using BigDL, utilizing 4 executors and training with a batch size of 32. Notably, Figure 7 shows that the training took place over the course of 9 minutes and required the use of 32 cores in total as each executor has 8 Cores. Spark monitoring instrumentation has been used to measure the executor CPU utilization by JVM.

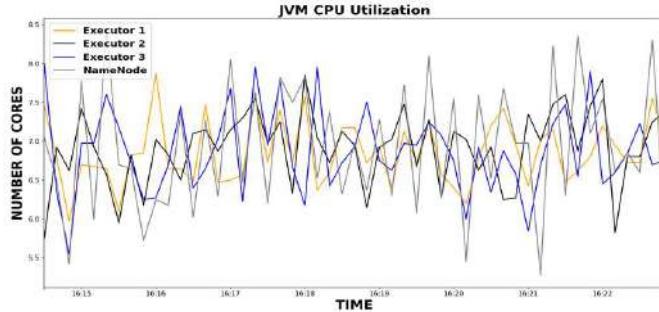


Figure 7: CPU utilization by JVM during training task

Model Training Performance: Figure 8 describes the training time of all the models in our experimentation. We can see in the case of the CNN (inception) model, the Colab and BigDL cluster perform close to 2 minutes but the local computer takes around 4 minutes. When compared to CNN, the time disparity between BigDL and Colab for RNN increases. It takes about 3 minutes in BigDL and about 5 minutes in Colab. In the local environment the model consumes most time. The BigDL and colab models for B-RNN take about 8 minutes, but the local PC takes almost 10 minutes. BigDL and colab differ in time by about two minutes in LSTM. In the case of an auto encoder and a feed-forward network, the time difference in various environments is comparable, but both models are much faster than LSTM. Specifically, the feed forward network is the fastest one. From the chart it is evident that our environment runs faster for all models except B-RNN. It is to be noted that for Local CPU, we used 5000×10 linear classifier structure by downscaling the images. Evaluations show that on average, Google Colab performs 15% and Local PC performs 72% less efficiently than our BigDL cluster for training.



Figure 8: Training time comparison of different Deep learning models in different architectures

Communication Overhead: Figure 9 describes the communication overhead in our test environments. The BigDL Cluster has the largest overhead due to the distributed nature of the data, there is more communication between the servers to retrieve data. In the graph, the BigDL cluster has more communication overhead compared to the Google Colab VM. Due to the Gbps LAN connection we have established and the fact that we have completed data transformation, the time it takes to transport 80 Gb of data is almost 10 seconds. The Local Machine setup has the least overhead ranging from 0.79 to 10 seconds because there is no outside communication delay with any other machine for data transfer.

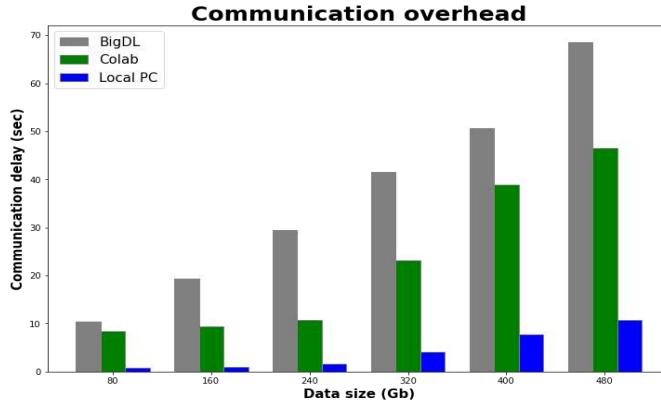


Figure 9: Communication overhead with different data sizes

5 CONCLUSIONS

For the proposed technique, we trained different deep learning models using our BigDL cluster. The cluster gave us very comparable performance to GPU based setups such as Google Colab or Kaggle Notebooks, even better in some models. This cluster provided almost twice the performance of a standalone computer. By using the automated setup GUI in our model, users can easily create clusters such as this through connecting devices using a LAN switch. This model is very useful for researchers and students who use computational resources such as GPUs. By using the Apache Spark and BigDL stack, users can also gain a huge boost in data throughput and transfer rates. As BigDL is still a relatively new framework, more features should be added to make it more compact. More research should be done to optimize factors such as Communication Overhead, Energy Consumption etc.

References

- [1] Fernández, A. M., Gutiérrez-Avilés, D., Troncoso, A., & Martínez-Álvarez, F. (2020). “Automated Deployment of a Spark Cluster with Machine Learning Algorithm Integration”. *Big Data Research*, 19, 100135.
- [2] Kim, H., Park, J., Jang, J., & Yoon, S. (2016). “Deepspark: Spark-based deep learning supporting asynchronous updates and caffe compatibility.”
- [3] Mostafaeipour, A., Jahangard Rafsanjani, A., Ahmadi, M., & Arockia Dhanraj, J. (2021). “Investigating the performance of Hadoop and Spark platforms on machine learning algorithms.” *The Journal of Supercomputing*, 77(2), 1273-1300.
- [4] Ghoting, Amol, Rajasekar Krishnamurthy, Edwin Pednault, Berthold Reinwald, Vikas Sindhwani, Shirish Tatikonda, Yuanyuan Tian, and Shivakumar Vaithyanathan. "SystemML: Declarative machine learning on MapReduce." In *2011 IEEE 27th International Conference on Data Engineering*, pp. 231-242. IEEE, 2011.
- [5] Jason Jinquan Dai, Yiheng Wang, Xin Qiu, Ding Ding, Yao Zhang, Yanzhang Wang, Xianyan Jia, Cherry Li Zhang, Yan Wan, Zhichao Li, Jiao Wang, Shengsheng Huang, Zhongyuan Wu, Yang Wang, Yuhao Yang, Bowen She, Dongjie Shi, Qi Lu, Kai Huang, and Guoqiong Song.

2019. "BigDL: A Distributed Deep Learning Framework for Big Data." In Proceedings of the ACM Symposium on Cloud Computing (SoCC '19).
- [6] Xiangrui Meng, Joseph Bradley, Burak Yavuz, Evan Sparks, Shivararam Venkataraman, Davies Liu, Jeremy Freeman, DB Tsai, Manish Amde, Sean Owen, Doris Xin, Reynold Xin, Michael J. Franklin, Reza Zadeh, Matei Zaharia, and Ameet Talwalkar. 2016. "MLlib: machine learning in apache spark." *J. Mach. Learn. Res.* 17, 1 (January 2016), 1235–1241.
 - [7] M. Langer, A. Hall, Z. He and W. Rahayu, "MPCA SGD—A Method for Distributed Training of Deep Learning Models on Spark," in *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 11, pp. 2540-2556, 1 Nov. 2018.
 - [8] Kim, H., Park, J., Jang, J., & Yoon, S. (2016). "DeepSpark: A Spark-Based Distributed Deep Learning Framework for Commodity Clusters."
 - [9] Z. Li, J. Davis and S. A. Jarvis, "Optimizing Machine Learning on Apache Spark in HPC Environments," 2018 IEEE/ACM Machine Learning in HPC Environments (MLHPC), 2018, pp. 95- 105.
 - [10] A. Khumoyun, Yun Cui and Lee Hanku, "Spark based distributed Deep Learning framework for Big Data applications," 2016 International Conference on Information Science and Communications Technologies (ICISCT), 2016, pp. 1-5.s
 - [11] Aspri, Maria & Tsagkatakis, Grigoris & Tsakalides, Panagiotis. (2020). "Distributed Training and Inference of Deep Learning Models for Multi-Modal Land Cover Classification. Remote Sensing."
 - [12] Nikitha Johnsirani Venkatesan, ChoonSung Nam & Dong Ryeol Shin (2018): "Deep Learning Frameworks on Apache Spark: A Review, IETE Technical Review."
 - [13] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
 - [14] Nguyen, T. Q., Weitekamp, D., Anderson, D., Castello, R., Cerri, O., Pierini, M., ... & Vlimant, J. R. (2019). "Topology classification with deep learning to improve real-time event selection at the LHC." *Computing and Software for Big Science*, 3(1), 1-14.
 - [15] Jonnalagadda, V. S., Srikanth, P., Thumati, K., Nallamala, S. H., & Dist, K. (2016). "A review study of apache spark in big data processing." *International Journal of Computer Science Trends and Technology (IJCST)*, 4(3), 93-98.
 - [16] Paul Fiterău-Broștean, Toon Lenaerts, Erik Poll, Joeri de Ruiter, Frits Vaandrager, and Patrick Verleg. 2017. "Model learning and model checking of SSH implementations." In *Proceedings of the 24th ACM SIGSOFT International SPIN Symposium on Model Checking of Software (SPIN 2017)*. Association for Computing Machinery, New York, NY, USA, 142–151.
 - [17] Dai, J. J., Wang, Y., Qiu, X., Ding, D., Zhang, Y., Wang, Y., ... & Song, G. (2019, November). "Bigdl: A distributed deep learning framework for big data." In *Proceedings of the ACM Symposium on Cloud Computing* (pp. 50-60)
 - [18] Aftab, M. O., Awan, M. J., Khalid, S., Javed, R., & Shabir, H. (2021, April). "Executing spark BigDL for leukemia detection from microscopic images using transfer learning." In *2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)* (pp. 216-220). IEEE
 - [19] Borthakur, D. (2008). "HDFS architecture guide." *Hadoop apache project*, 53(1-13), 2
 - [20] Jain, M. (2018). "Advanced Techniques in Shell Scripting." In *Beginning Modern Unix* (pp. 283-312). Apress, Berkeley, CA.
 - [21] Liashchynskyi, P., & Liashchynskyi, P. (2019). "Grid search, random search, genetic algorithm: a big comparison for NAS."

Blockchain-Driven Cloud Service: A Survey

*Hamed Taherdoost¹

¹Departement of Arts, Communications and Social Sciences, University Canada
West (UCW), Vancouver, Canada
¹*hamed.taherdoost@gmail.com

* Hamed Taherdoost

Abstract. Cloud services run centrally and store data before making it available to customers through software, and use large servers. When employing these centralized-based methods, organizations often have to make compromises in the areas of authorization, privacy, and security. Due to the fact that once data is recorded, it cannot be amended without also affecting the data in all preceding blocks, blockchains may restrict the modification of any data in any of the blocks. The key benefit of integrating blockchain into cloud services is increased data security. When cloud services and blockchain are merged, it's like having the best of both worlds in one convenient package. Cloud services powered by blockchain will be covered in this survey, along with their uses, difficulties, and prospects for the future.

Keywords: Blockchain, Cloud Service, Cloud Computing, Security, Privacy

1 Introduction

Large servers are used in cloud services, which operate in a centralized manner and store data before making it accessible to users via software [1]. Organizations often have to compromise on authorization, privacy, and security when using these centralized-based approaches [2]. Because the information stored may be accessed by anyone, even unlawfully by hackers and viruses, cloud services often lack trust and frequently results in the data leakage of particularly sensitive information [3]. Cloud architecture is how technological components combine to form a cloud, in which resources are pooled and shared through virtualization technologies and a network [4].

A blockchain is an ever-expanding, linked list-like structure of data that comprises blocks connected by links and saved via encryption [5]. Each "block" on a blockchain has a timestamp, a cryptographic hash of the prior block, and data relevant to a specific transaction [5, 6]. Since each block contains a link to and information about its predecessor, the previous block cannot be edited or destroyed [7]. Once data is stored in a block, it cannot be modified without also modifying the data in every block before it [8].

Increasing data security is one of the main advantages of using blockchain technology in cloud services [9, 10]. It's like getting the best of both worlds in a single package when blockchain technology and cloud services are combined. This survey will include cloud services powered by blockchain from applications to challenges and future perspectives.

2 Using Blockchain Technology in Cloud Services: The Range of Applications

Due to its unique characteristics, the blockchain system might significantly enhance functionality or performance. As a decentralized system, blockchain could be of great assistance in developing an architecture in which multiple computers can operate concurrently on a single task, such as data processing or storage, thereby reducing the total duration of the operation and accelerating data processing and uploading [11].

In addition, the security of cloud services has increased. As cloud services often manage vast quantities of data, there is always a danger of data compromise. In addition, since cloud services depend on a centralized design, hackers may breach the central server, causing the whole system to fail with no backup of any lost data. Therefore, there is the possibility of implementing blockchain in cloud services to overcome these issues (Fig. 1). Table 1 provides examples of some relevant research that were released in 2022. In the following domains, blockchain can surpass existing cloud solutions:

- **Higher Data Security:** Internet of Things (IoT) data kept in the cloud is frequently associated with the homeowner's personal information, including personal habits, real estate, household products, voice recordings, and video footage. Leaks of sensitive data may compromise personal security by facilitating thefts, assaults, and the illicit selling of personal information for monetary gain [12]. In light of these conditions, the cloud infrastructure is in jeopardy. The solution to this problem is the use of blockchain in cloud services, which may increase the security of the whole architecture.
- **Scalability:** Large-scale blockchain applications may subject blockchain networks to extraordinarily high transaction volumes. To offer scalable blockchain systems, it is essential to have robust data processing services with fast transaction execution speeds. In this sector, the cloud may deliver on-demand services to blockchain operations. due to its scalability possibilities. In other words, integrating blockchain with cloud services may result in a very scalable solution.
- **More Effective Goods and Service Ownership Tracking:** The logistics industry has a basic challenge in regularly keeping track of all the vehicles in its network, their current locations, the amount of time each vehicle spends at a certain location, and how to establish communication between many vehicles. Similar to complete

testing for packages, software products face challenges due to their centralized architecture. Blockchain has considerable potential for monitoring these goods and services.

- **Decentralization:** The dependence on a single server for managing data and decision-making is a basic concern with cloud services and IoT. A central server failure, for instance, could disrupt the whole system and lead to the loss of vital data stored on the server. Additionally, the primary server is subject to hacker assaults. The blockchain may offer a solution to this problem since it employs a decentralized design that precludes the failure of the whole system if a single server goes down by keeping several copies of identical data on many computer nodes. Additionally, data loss is not a concern since there are several versions of the data on separate nodes.
- **Tolerance for Faults:** It may help with fault tolerance if blockchain data is replicated in the cloud over a network of computers that are tightly connected through collaborative clouds. As a consequence, there will be less chance of a single failure brought on by the disruption of any cloud node, allowing for uninterrupted services.

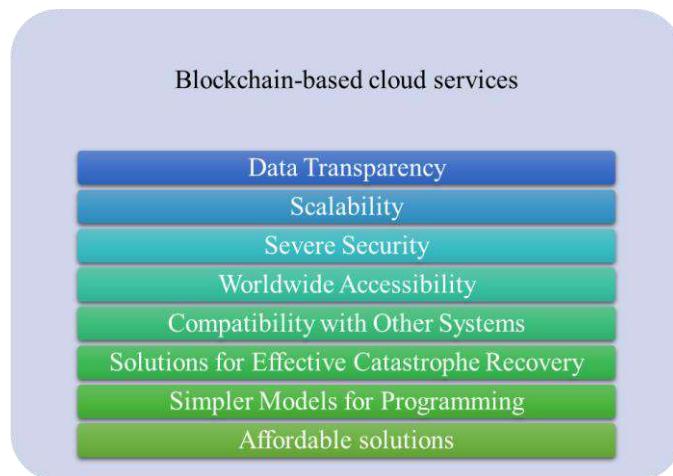


Fig .1. Benefits of blockchain-powered cloud services

Table 1. Some related studies were published in 2022

System	Feature	Reference
Blockchain-based multi-cloud searchable encryption scheme	High performance and security	[13]

Platform for integrated cloud services in the energy market	Energy trading's simplicity and security, as well as energy delivery's effectiveness and intelligence	[14]
Novel service level agreement paradigm based on blockchain and smart contracts	Data security	[15]
Blockchain- and cloud service-based smart home	Good applicability	[16]

3 Challenges and Future Direction

The cloud services model varies from centralized data centers in terms of security issues. The cloud offers some advantages over conventional data centers, but it also has several disadvantages, including user administration and device management. Combining blockchain technology with the cloud services idea may increase cloud services security. The cloud platform improves scalability and extensive analysis. Cloud service might become more dependable and safe thanks to blockchain technology. The computational power and storage capabilities of cloud services can meet the needs of the blockchain. In the context of cloud services, several blockchain security services are being categorized and debated. Blockchain and cloud services may combine their strengths to get beyond the most frequent obstacles. Study limitations include high hardware needs and extreme energy usage. Additionally, the transaction takes up a lot of time. This is a serious issue that has to be addressed. The mining industry may benefit from increased revenue due to more efficient currency creation, but blockchain adoption may also accelerate as a result. Many businesses may reduce their carbon footprints by integrating blockchain into all areas of the economy.

In today's business context, a heavy reliance on the cloud and associated risks may be risky and harmful [17]. The cloud's centralized, compliant, and secure nature poses a serious risk. Blockchain technology is used to enhance data storage security, transaction speed, reliability, and corporate processes. One of the best courses of action is to combine blockchain with cloud services since it offers more security and decentralization while enabling better authorization and privacy [18]. As the blockchain confronts new problems in the future, such as more security, data storage at every node, etc., it will be required to keep working on these challenges [19]. When it comes to protecting their data, which is essential to their capacity to function and make decisions in today's world of constant competition, organizations may benefit from research on how to employ this combination.

4 Conclusion

In conclusion, blockchain may be very advantageous for cloud services. It is a useful technique for providing security and privacy in a cloud setting. Thanks to technology, the cost of acquiring cloud resources might be reduced. It also offers a platform for individuals to develop trust with one another. This poll addressed blockchain-powered cloud services, as well as their benefits for better data security, scalability, tracking the ownership of products and services, decentralization, and fault tolerance. Various blockchain security services are being classified and discussed in the context of cloud services. Blockchain technology and cloud services could combine to solve the most frequent problems. It will be necessary to continue working on these difficulties when the blockchain faces additional issues in the future, such as increased security, data storage at every node, etc. Organizations may profit from a study on how to use this combination when it comes to preserving their data, which is crucial to their ability to operate and make choices in today's world of continual competition.

References

1. Shawish, A. and M. Salama, *Cloud computing: paradigms and technologies*, in *Inter-cooperative collective intelligence: Techniques and applications*. 2014, Springer. p. 39-67.
2. Taherdoost, H., *Understanding Cybersecurity Frameworks and Information Security Standards—A Review and Comprehensive Overview*. Electronics, 2022. **11**(14): p. 2181.
3. Barona, R. and E.M. Anita. *A survey on data breach challenges in cloud computing security: Issues and threats*. in *2017 International conference on circuit, power and computing technologies (ICCPCT)*. 2017. IEEE.
4. Ibrahim, A.S., J. Hamlyn-Harris, and J. Grundy, *Emerging security challenges of cloud virtual infrastructure*. arXiv preprint arXiv:1612.09059, 2016.
5. Pahlajani, S., A. Kshirsagar, and V. Pachghare. *Survey on private blockchain consensus algorithms*. in *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*. 2019. IEEE.
6. Krishnapriya, S. and G. Sarath, *Securing land registration using blockchain*. Procedia Computer Science, 2020. **171**: p. 1708-1715.
7. Taherdoost, H., *Blockchain Technology and Artificial Intelligence Together: A Critical Review on Applications*. Applied Sciences 2022. **12**(24): p. 12948.
8. Hanifatunnisa, R. and B. Rahardjo. *Blockchain based e-voting recording system design*. in *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*. 2017. IEEE.
9. Esposito, C., et al., *Blockchain: A panacea for healthcare cloud-based data security and privacy?* IEEE Cloud Computing, 2018. **5**(1): p. 31-37.
10. Taherdoost, H., *A Critical Review of Blockchain Acceptance Models—Blockchain Technology Adoption Frameworks and Applications*. Computers, 2022. **11**(2): p. 24.
11. Moosavi, N. and H. Taherdoost. *Blockchain-Enabled Network for 6G Wireless Communication Systems*. in *International Conference on Intelligent Cyber Physical Systems*

- and Internet of Things (ICoICI 2022)*. 2022. Coimbatore, India: Engineering Cyber-Physical Systems and Critical Infrastructures, Springer.
- 12. Moosavi, N. and H. Taherdoost, *Blockchain and Internet of Things (IoT): A Disruptive Integration*, in *2nd International Conference on Emerging Technologies and Intelligent Systems (ICETIS 2022)*. 2022, Lecture Notes in Networks and Systems, Springer: virtual conference.
 - 13. Fu, S., C. Zhang, and W. Ao, *Searchable encryption scheme for multiple cloud storage using double-layer blockchain*. Concurrency and Computation: Practice and Experience, 2022. **34**(16): p. e5860.
 - 14. Wang, L., et al., *Design of integrated energy market cloud service platform based on blockchain smart contract*. International Journal of Electrical Power & Energy Systems, 2022. **135**: p. 107515.
 - 15. Tan, W., et al., *A novel service level agreement model using blockchain and smart contract for cloud manufacturing in industry 4.0*. Enterprise Information Systems, 2022. **16**(12): p. 1939426.
 - 16. Liao, K., *Design of the Secure Smart Home System Based on the Blockchain and Cloud Service*. Wireless Communications and Mobile Computing, 2022. **2022**.
 - 17. Taherdoost, H., *A review on risk management in information systems: Risk policy, control and fraud detection*. Electronics, 2021. **10**(24): p. 3065.
 - 18. Gong, J. and N.J. Navimipour, *An in-depth and systematic literature review on the blockchain-based approaches for cloud computing*. Cluster Computing, 2021: p. 1-18.
 - 19. Peng, L., et al., *Privacy preservation in permissionless blockchain: A survey*. Digital Communications and Networks, 2021. **7**(3): p. 295-307.

Interaction Analysis of a Multivariable Process in A Manufacturing Industry – a review

P k juneja^{1*}, T Anand², P Saini^{3*}, Rajeev Gupta⁴, F S Gill⁵, S Sunori⁶

¹Department of Electronics and Communication Engineering

²Department of PDP

³Department of Electrical Engineering

⁴Department of Electronics and Communication Engineering

⁵Department of Physics

⁶Department of Electronics and Communication Engineering

^{1,2,3,5}Graphic Era Deemed to be University, Dehradun, Uttarakhand, India

⁴Graphic Era Hill University, Dehradun, Uttarakhand, India

⁶Graphic Era Hill University, Bhimtal, Uttarakhand, India

Email id: mailjuneja@gmail.com, parvesh.saini.eee@gmail.com

* Corresponding Author

Abstract. It is important to perform interaction analysis in case of Multi-Variable (MV) control system to determine the best loop pairing recommendation. Non desirable couplings can be decreased with the help of incorporation of suitable decouplers designed but only after determining suitable loop pairing. The present work extensively reviews the literature survey pertaining to the interaction analysis and loop pairing recommendation in a multivariable control system. Multivariable interaction techniques like Relative Gain Array (RGA), Effective RGA (ERGA), Dynamic RGA (DRGA), input relative gain array (IRGA), Normalized RGA (NRGA), Normalized Effective RGA (NERGA), Relative Omega Array (ROmA), etc. and their applications in various industrial processes are highlighted in brief.

Keywords: MIMO, RGA, interaction analysis, loop pairing, process control

1. Introduction

Process industries mostly have multivariable process along with undesirable cross couplings. The performance of the Multi-Input Multi-Outpu (MIMO) process is affected because of these interactions. A good process control is required in these process industries to increase product rate and enhance product quality along with less energy consumption and less release of pollutant by-products in the form of flue gases.

This requires proficient design of equipment, process optimization, application of recovery and recycling techniques and appropriate control system design. A

typical MIMO system is represented in the figure 1.

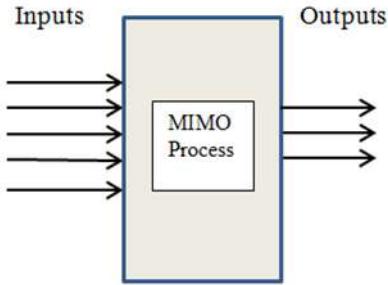


Fig.1. Block representation of a MIMO system [1]

2. Literature Review

J. Lieslehto and H. N. Koivo [1] demonstrated the expert system pertaining to loop pairing analysis of Linear Multi Variable (MV) process. W. Feng and M. J. Grimble [2] proposed a new performance index for transient and steady state that improves stability and robustness of the selected linear MV system. X. Luan et al. [3] investigated RNGA index for complex MV systems subjected to step and ramp inputs. Also, average residence time is calculated for complex process models.

P. Samuelsson et al. [4] compared RGA and HIIA for finding interactions in a wastewater treatment plant. Q. -F. Liao et al. [5] presented a loop pairing method for multivariable process which provide more accuracy in comparison to RNGA method. C Le Brun et al. [6] implemented modelling of turboprop engine and applied decentralized control design with full compensator on it.

B Wang et al. [7] obtained the coupling degree influenced by ac induction motor speed and demonstrated that with the increase of the motor speed, the degree of coupling will increase too. A. Khaki-Sedigh and B. Moaveni [8] proposed a test for identification of variation in input-output pairing under parametric uncertainties' presence. H Jiang et al. [9] presented DRGA technique based on state feedback predictive control (SFPC).

W. Liang et al. [10] developed a DRGA method to select an input for a particular output in Model Order Reduction (MOR) process. In DRGA method steady state and transient responses are involved in the process of loop pairing. In case of RLC networks, DRGA based MOR technique is more accurate in comparison to classical MOR methods. J. Hofmann et al. [11] considered a biomass pyrolysis process and used RGA, DRGA and Singular Value Analysis (SVA) for interaction

analysis for system with unmeasurable disturbances or dead time. J. Chen et al. [12] studied deviations in properties (open-loop) under modelling uncertainties. An example is constructed to show that it is difficult to control plants having large values of CN (Condition Number) and RGA.

Y Zhang et al. [13] introduced an IRGA to the design of Tornambe's Controller to find a proper match between the inputs and outputs. K. C et al. [14] considered a four-tank system and applied RGA method and decoupling control. W. Hu et al. [15] proposed a technique to overcome limitations of classical techniques for a multivariable process containing differentiators or integrators or both.

G. He et al. [16] studied coupling in a maglev bogie using ERGA and validated it experimentally. C. -C. Tsai et al. [17] presented a predictive Proportional Integrator temperature control using PSO technique for molding modules in semiconductor die packaging machines. M. A. Moezzi et al. [18] introduced NERGA and applied in an adaptive decentralized PID control technique.

P Juneja et al. [19] presented a systematic steps pertaining to multivariable control system analysis through the flow chart. I. Muntean et al. [20] shown that applying RGA analysis for a 1st principle model, a decentralized control strategy using PI controllers can be easily developed. A. Fatehi and A. Shariati [21] introduced NRGA matrix and emphasized that pairing problem is similar to an assignment problem.

A. Avvalabadi and A. Shahmansourian [22] presented a new easy to interpret strategy which gives suitable pairing recommendations. S. Choobkar et al. [23] applied minimum variance (MV) index for loop pairing. A. Balestrino and A. Landi [24] presented ROmA which retains the simplicity of RGA but include dynamic interactions.

L. Braccia and D. Zumoffen [25] suggested technique which utilizes RGA approach within a deterministic MILP formulation. E H Bristol et al. [26] Presented results concerning dynamic effects, relating decoupling and optimal control. J Hofman et al. [27] applied participation matrix for pairing recommendation of the falling evaporator process. A. Balestrino et al. [28] addressed the problem of pairing for a nonlinear and timevarying system. ARGA is proposed for solving both integrity problems.

Table 1. Literature review

S. No.	Authors and year	System selected/	analysis
--------	------------------	------------------	----------

developed			
1	J. Lieslehto and H. N. Koivo, 1987	LTIVMV	IA
2	W. Feng and M. J. Grimble, 1989	Ss response and transient response of stable and robust system.	Proposed performance interaction measure
3	X. Luan, Q. Chen and F. Liu, 2017	non-square processes	RNGA Relative normalized gain array
4	P. Samuelsson et al., 2005	wastewater treatment plant	RGA and HIIA
5	Q. -F. Liao et al., 2013	type-2 T-S fuzzy models	RNGA
6	C. Le Brun et al., 2015	Turboprop engine	Coupling analysis and Decentralized control
7	B. Wang, Q. Zhu, H. Zang, S. Kou and J. Ke, 2018	AC induction motor	Coupling degree influence by motor speed calculation
8	A. Khaki-Sedigh and B. Moaveni, 2003	uncertain multivariable plants	RGA
9	H. Jiang et al., 2012	MV Case study	DRGA
10	W. Liang et al., 2020	RLC networks	DRGA, MOR
11	J. Hofmann et al., 2019	Biomass Pyrolysis Process	RGA, DRGA, SVA
12	J. Chen et al., 1992	Example model	RGA, Condition Number
13	Y. Zhang et al., 2011	Tornambe's Controller	IRGA
14	K. C, D. S, T. S. K, A. K and V. R, 2021	Four-tank system	RGA IMC PSO based decoupling SCADA
15	W. Hu et al., 2010	MIMO process containing integrators and/or differentiators	Proposed new method
16	G. He et al., 2013	maglev bogie linearized model	ERGA
17	C. C. Tsai et al., 2017	Molding process in semiconductor die packaging machine	PSO-RGA
18	M. A. Moezzi et al.	High dimensional system	NERGA
19	P. Juneja et al.	MV system	RGA, NI, MRI
20	I. Muntean et al., 2015	wastewater treatment plant	RGA

21	A. Fatehi and A. Shariati	MIMO plant	NRGA
22	A. Avvalabadi and A. Shahmansourian	Several examples	Proposed method
23	S. Choobkar et al.	MV system	minimum variance (MV) index
24	A. Balestrino and A. Landi	examples	ROmA
25	L. Braccia and D. Zumoffen	Industrial process	RGA, Mixed Integer Linear Programming RGA
26	E H Bristol, 1977	examples	
27	J.Hofmann et al., 2021	Falling Film Evaporator Process,	participation matrix and HIIA
28	A. Balestrino et al., 2008	examples	Absolute RGA

3. Conclusion

Present literature review covers the important aspects in any manufacturing industry relevant to interaction analysis of various variables in a multivariable process. Review is exhaustive in nature and it not only covers relative gain array (the commonly used interaction measuring index) but includes all the updated, hybrid, modified or proposed interaction index.

References

- 1 J. Lieslehto and H. N. Koivo, "An Expert System for Interaction Analysis of Multivariable Systems," 1987 American Control Conference, 1987, pp. 961-962.
- 2 W. Feng and M. J. Grimble, "A New Procedure to Account for Performance Interaction in Multivariable Systems," 1989 American Control Conference, 1989, pp. 1280-1285, doi: 10.23919/ACC.1989.4790385.
- 3 X. Luan, Q. Chen and F. Liu, "Interaction measurement for complex multivariable models with various reference inputs based on RNGA," 2017 11th Asian Control Conference (ASCC), 2017, pp. 2411-2416, doi: 10.1109/ASCC.2017.8287552.
- 4 P. Samuelsson, B. Halvarsson and B. Carlsson, "Interaction analysis and control structure selection in a wastewater treatment plant model," in IEEE Transactions on Control Systems Technology, vol. 13, no. 6, pp. 955-964, Nov. 2005, doi: 10.1109/TCST.2005.854322.
- 5 Q. -F. Liao, W. -J. Cai and Y. -Y. Wang, "Interaction analysis and loop pairing for MIMO processes described by type-2 T-S fuzzy models," 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA), 2013, pp. 622-627, doi: 10.1109/ICIEA.2013.6566443.

- 6 C. Le Brun, E. Godoy, D. Beauvois, B. Liacu and R. Noguera, "Coupling analysis and control of a turboprop engine," 2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO), 2015, pp. 420-427.
- 7 B. Wang, Q. Zhu, H. Zang, S. Kou and J. Ke, "Coupling Analysis of AC Motors Based on Relative Gain Array," 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2018, pp. 2473-2476, doi: 10.1109/IAEAC.2018.8577826.
- 8 A. Khaki-Sedigh and B. Moaveni, "Relative gain array analysis of uncertain multivariable plants," 2003 European Control Conference (ECC), 2003, pp. 2862-2866, doi: 10.23919/ECC.2003.7086474.
- 9 H. Jiang, M. Wei, F. Xu and X. Luo, "A dynamic relative gain array based on model predictive control," Proceedings of the 10th World Congress on Intelligent Control and Automation, 2012, pp. 3340-3344, doi: 10.1109/WCICA.2012.6358450.
- 10 W. Liang, H. -B. Chen, G. He and J. Chen, "Model Order Reduction Based on Dynamic Relative Gain Array for MIMO Systems," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 67, no. 11, pp. 2507-2511, Nov. 2020, doi: 10.1109/TCSII.2019.2962709
- 11 J. Hofmann, H. -C. Holz and L. Gröll, "Relative Gain Array and Singular Value Analysis to Improve the Control in a Biomass Pyrolysis Process," 2019 IEEE 15th International Conference on Control and Automation (ICCA), 2019, pp. 596-603, doi: 10.1109/ICCA.2019.8900025.
- 12 J. Chen, J. S. Freudenberg and C. N. Nett, "On relative gain array and condition number," [1992] Proceedings of the 31st IEEE Conference on Decision and Control, 1992, pp. 219-224 vol.1, doi: 10.1109/CDC.1992.371751.
- 13 Y. Zhang, D. Li and D. Lao, "IRGA for MIMO systems based on TC," Proceedings of the 30th Chinese Control Conference, 2011, pp. 3803-3808.
- 14 K. C, D. S, T. S. K, A. K and V. R, "RGA Analysis and Hardware Implementation of PSO Based Decoupled Mimo System," 2021 4th International Conference on Computing and Communications Technologies (ICCCT), 2021, pp. 448-452, doi: 10.1109/ICCCT53315.2021.9711818.
- 15 W. Hu, W. -J. Cai and G. Xiao, "Relative gain array for MIMO processes containing integrators and/or differentiators," 2010 11th International Conference on Control Automation Robotics & Vision, 2010, pp. 231-235, doi: 10.1109/ICARCV.2010.5707220.
- 16 G. He, J. Li, Y. Li and P. Cui, "Interactions analysis in the maglev bogie with decentralized controllers using an effective relative gain array measure," 2013 10th IEEE International Conference on Control and Automation (ICCA), 2013, pp. 1070-1075, doi: 10.1109/ICCA.2013.6564922.
- 17 C. -C. Tsai, F. -C. Tai and R. -S. Liu, "Intelligent predictive temperature control using PSO-RGA for transfer mold heating processes in semiconductor die packaging machines," 2017 International Conference on System Science and Engineering (ICSSE), 2017, pp. 510-514, doi: 10.1109/ICSSE.2017.8030926.
- 18 M. . -A. Moezzi, A. Fatehi and M. A. Nekoui, "A novel automatic method for multivariable process pairing and control," 2008 Annual IEEE India Conference, 2008, pp. 262-267, doi: 10.1109/INDCON.2008.4768837.
- 19 P. Juneja, A. Sharma, V. Joshi, H. Pathak, S. K. Sunari and A. Sharma, "Delayed Complex Multivariable Constrained Process Control – A Review," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2020, pp. 569-573, doi: 10.1109/ICACCCN51052.2020.9362800.
- 20 I. Muntean, R. Both, R. Crisan and I. Nascu, "RGA analysis and decentralized control for a wastewater treatment plant," 2015 IEEE International Conference on Industrial Technology (ICIT), 2015, pp. 453-458, doi: 10.1109/ICIT.2015.7125140.

- 21 A. Fatehi and A. Shariati, "Automatic pairing of MIMO plants using normalized RGA," 2007 Mediterranean Conference on Control & Automation, 2007, pp. 1-6, doi: 10.1109/MED.2007.4433929.
- 22 A. Avvalabadi and A. Shahmansourian, "A new mathematical approach for input-output pairing in MIMO square systems," 2014 22nd Iranian Conference on Electrical Engineering (ICEE), 2014, pp. 1244-1247, doi: 10.1109/IranianCEE.2014.6999725.
- 23 S. Choobkar, A. Khaki Sedigh and A. Fatehi, "Input-output pairing based on the control performance assessment index," 2010 2nd International Conference on Advanced Computer Control, 2010, pp. 492-496, doi: 10.1109/ICACC.2010.5486691.
- 24 A. Balestrino and A. Landi, "ROmA loop pairing criteria for multivariable processes," 2007 European Control Conference (ECC), 2007, pp. 4765-4771, doi: 10.23919/ECC.2007.7068387.
- 25 L. Braccia and D. Zumoffen, "The input-output pairing problem: An optimization based approach," 2015 XVI Workshop on Information Processing and Control (RPIC), 2015, pp. 1-6, doi: 10.1109/RPIC.2015.7497058.
- 26 E. H. Bristol, "RGA 1977: Dynamic effects of interaction," 1977 IEEE Conference on Decision and Control including the 16th Symposium on Adaptive Processes and A Special Symposium on Fuzzy Set Theory and Applications, 1977, pp. 1096-1100, doi: 10.1109/CDC.1977.271734.
- 27 J. Hofmann, L. Gröll and V. Hagenmeyer, "Control Loop Pairing and Interaction Analyses of the Falling Film Evaporator Process," 2021 23rd International Conference on Process Control (PC), 2021, pp. 186-193, doi: 10.1109/PC52310.2021.9447487.
- 28 A. Balestrino, E. Crisostomi, A. Landi and A. Menicagli, "ARGA loop pairing criteria for multivariable systems," 2008 47th IEEE Conference on Decision and Control, 2008, pp. 5668-5673, doi: 10.1109/CDC.2008.4738840.

Machine Learning Approaches for Educational Data Mining

Mr. Mahesh Bapusaheb Toradmal¹,Dr. Mita Mehta^{2[0000-0001-9876-0136]} and Dr Smita Mehendale³

^{1,2}Symbiosis Institute of Business Management, Symbiosis International (Deemed University), Pune, India

³Symbiosis Institute of Management Studies, Symbiosis International (Deemed University), Pune, India

mitamehta@sibmpune.edu.in

Abstract. Educational Data Mining (EDM) has enhanced one of the essential fields nowadays because, with technology improvement, students' difficulties are also expanding. To grab these difficulties and encourage students, educational data mining has come into continuation. Educational Data Mining is the process to evaluate the student's academic performance. Learning analytics apply machine learning techniques to have better and accurate interpretation out of it. Several researchers used it as a prediction system to predict student performance. This article focuses on various techniques used in EDM for classification and analysing of this data to build strong recommendation system. Numerous machine learning algorithm has used in different existing systems and predict the classification accordingly. Moreover, it also analysed the challenges identified when EDM deals with large data sets. In this discussion, we evaluate the few algorithms and tried to propose new methodology to generate better classification and recommendation.

Keywords: Educational Data Mining, Web Mining, Machine Learning, Data Pre- Processing, Feature Extraction.

1. Introduction

Education data mining as a research topic is still 'young'. Past studies have shown the significant potential in the topic from various parts of the world. Various aspects of education related gaps like curriculum design, time management of the course, students behavioural and learning pattern can be analysed using educational data mining concept. Every change in educational environment and in its delivery, methods is focused on providing better education quality to learners. Appropriate knowledge is required for teaching institution to evaluate and enhance the learning process. Students are considered the critical assets rather than liability of any institution. Student is the product which institutes produce. The role of measuring and forecasting students' potential to succeed in life is part of the quality improvement of learning processes.

The unpredictable processing of academic knowledge is one of the enormous absolutes of higher educational frameworks. Every higher educational institution's primary aim is to improve the essence of organizational decisions and provide training programs. One path to achieving the highest profit in the advanced education system is a reasonable forecast of understudy's accomplishment in secondary education organizations. It is of extraordinary talent to the higher learning administrators to predict the execution of an assistant. It is a testing activity to predict the implementation of both the accomplices.

Specialized schools are productive because they have dignified understudies and employees and a measure certification system that continuously improves them.

In the open-source software Weka 3.8 setting, this work evaluates EDM classification on different synthetic datasets. The data mining method of information disclosure is carried out. As the Weka tool Setting for Information Analysis, WEKA is extended. For online information mining operations, it is an aggregation of machine learning necessary calculations. System can either connect the calculations based straight-forward to a data set or call them from a Java code. The WEKA tool contains functions such as Data Pre-processing, Grouping, Regression, Clustering, Correlation, and The WEKA tool contains functions such as Data Pre-processing, Grouping, Regression, Clustering, Correlation and.

2. Literature Review

There are already various attempts by academics and authorities to mine students' information, hence the need to use the dimensionality of data in educational repositories for improved delivery of education institutions. Improvement is continuous process and for which we require continuous data mining as culture to see changes in trend of how students learn things and gain information. Purpose of this study is to study existing data mining methods used for EDM and see which one we could adopt for our field study further.

M Krishna et. al. [1] proposed tree logistic regression method of EDM with the help of statistics from a Moodle-based cohesive learning course to develop a module for the students which could predict performance. It uses credentials and Regression Trees (CART) decision [12] tree algorithm to prediction students at risk in early stage. This is based on the influence of four online actions and behaviours like sharing of message, development of some web-based content, opening course files and conducting some quiz online. In other study by

Juan L. et. al. [4] where author used machine learning methods and approach as one of the EDM technique to predict student's success based on raw data collected. Supervised learning techniques were used to predict student's behaviour to generate more precise and reliable results. Unambiguously, the authors used the Support Vector Machine (SVM) algorithm to produce reliable predictions. Apart from SVM other algorithms like Decision Tree (DT), Naïve Bytes (NB) and Random Forest (RF) were also well explored in this study.

Different techniques used in EDM already helped to improve student experience. One of the studies conducted by Ling Cen et. al. [2] in which award-winning mobile education program developed as outcome. program It was designed for the students studying in high school for their Chemistry subject. This introduced new know- how to help the students understand complex concepts in chemistry like molecular studies in better way. Use of interactive 3D simulation and visuals are used to improve learning experience of students while studying chemical reactions. This is also one kind of outcome of how we could use educational data to build better learning experience for students. Another study carried out by Prosanta Kumar Chaki et. al. [3] in which data collected from student's inputs for improvement of teaching strategy from specific teachers. Data was more focused on teacher's teaching attitude and skills. This was conducted to improve decision making quality improvement of teaching. They proposed automated decision-making framework to improve teaching strategy.

Study was conducted by Maria Tsiakmaki et. al. [5] around students from higher education level. In this study, author used deep neural networks to examine the learning of student's success. This study is important step in the field of EDM. It is reusing the pre trained model for new tasks. We call it as Transfer Learning (TR) in machine learning methods.

Sapna Arorra et. al. [6] in her study used both liner and logistic regression as EDM techniques to predict educational performance. This study performed two distinct investigations using these regressions. This study was conducted on the data collected from Delhi university college students.

This was real time data set to predict the performance of the individual students. Proposed system in this study acquires 71% and 85% classification accuracy.

Ierin Babu et. al. [7] This study used higher secondary score, intellectual quotient (IQ) and emotional quotient (EQ) levels of students as a data to predict their academic success by using modern algorithms. Experiment setup was asking students to enter their marks in system and appear for IQ & EQ tests with certain psychologist's guidance. Prediction model was formed in binary format. It predicts either student will pass or fail.

Khafidurrohman Agustianto and Prawidya Destarianto [8] This study focused on class misbalancing problem and proposed some balancing methods. This study driven on sampling sizes like the cycle of balancing is divided into three methods: under sampling, oversampling, and oversampling and under sampling combination. The method also focuses on integrating the Educational Data using the Neighbourhood Cleaning Code (NCL) under sampling methodology to achieve correct student modelling. Data that are under sampled using NCL was then categorized using the C4.5 algorithm for the Decision Tree.

Samuel-Soma M. Ajibade et. al. [9] heuristic function used in this study for student performance assessment. Purpose of study is to test the performance of students using a heuristic technique called as Differential Evolution for feature selection algorithms. In this participant dataset used along with other feature selection algorithms that were never used before. Classification methods such as NB, DT, K-Nearest Neighbor (KNN) and Discriminant Analysis (DISC) have also been used for evaluation purpose in this study as EDM techniques.

Khalid Al-Omar et. al. [10] explore the usability of the LMS at King Abdulaziz University using certain predefined usability criteria, such as a questionnaire on device usability scale (SUS). Another research used data mining techniques (sentiment analysis and clustering) to find out the specifics of any usability problems.

Ryan S. Baker [11] this researched done research in the field of EDM since from 1995. In his study he explained different challenges in it. This article discusses about how to eliminate the bugs from traditional data mining approaches.

These all studies conducted gives idea that there are many ways of implementing machine learning tools and techniques in EDM to predict the dependent variable under study. Precision and accuracy of prediction is driven by how we train the module and use the test data.

3. METHODOLOGIES

Literature review conducted shown that wide variety of EDM techniques has appeared over the last numerous years. Unusual is like these observed in data mining in different regions, whereas others are unusual to informative data mining. The four essential aspects of processes that are in especially repeatedly use by the EDM are forecast patterns, Composition Discovery, Relationship Mining, and Discovery with Standards. EDM's imperative factors are informative stakeholders, data mining mechanisms and techniques, educational goals, acquiring setting and purpose, and how these circumstances help realize full educational potential.

3.1 Classification: It is the multiple usually applied data mining method. It is the process of supervised learning that drafts data into various predefined classes. The analysis notions have been used for divining student attainment, achievement, experience, prophesying student failure, and identifying uncertain student's performance in EDM.

.

3.2 Clustering: It is the naming or classification of object groups that are related. It aims to display large databases to create useful assumptions for decision-making in the creation of other relationships, correlations, or groups. The use of educational segmentation is primarily intended to encourage the engagement of students in various learning circumstances, to suggest services and programs to users in a similar manner.

3.3 Statistics: Statistics is a mathematical framework that aims to use statistical methods to capture, analyses, interpret, or demonstrates graphical results. It could be used to determine different learning patterns required to direct teaching style creation using data from the web existing owners for instructional design.

3.4 Prediction: It is a highly beneficial in pushing forward in the educational system with new technologies. It is a tool that forecasts, instead of a present state, a potential state. This approach helps determine the success rate, fall-out, and maximum support of students.

3.5 Association Rule Mining: The association rule is a unique process for a specific input vector to explore relationships among variables and reference classes. In addition to making the courseware most effective, it is used to discover the type of smart based on students' features and knowledge and skills. This, due to it, helps teachers evaluate students' learning habits and more effectively arrange the course content. It can also be used to facilitate active learning, in addition to.

3.6 Regression: Regression is a forecasting tool used to evaluate the associations between the predictor variable (specified field) and one or more predictor variables and decide how such associations can lead to entities' educational goals.

3.7 Sequential Pattern: The concurrent pattern is a data mining method used to classify the associations among simultaneous occurrences, primarily to find some order throughout such incidents.

3.8 Text Mining: In various forms of both the web-based education system, this technique was successfully applied, often in group work, to provide automated process assessment that is supposed to carry out even in comment threads. Machine learning may increase educators' ability to evaluate the progress of both the group discussion and promote the process of generating instructional materials spent on user forum conversations.

3.9 Correlation: Correlation extraction is a test of the association of the two vectors of variables. It would be used in the educational system to estimate the success of students with regards to their final examination score and experiences in online homework mentoring, predict student progress in a university, define the main formative evaluation rules according to a specific learner's web-based learning portfolios and construct designers will tell that provide a more realistic image of prerequisite skills.

3.10 Outlier detection: Outlier testing is a process used from a broad dataset and identifies the intended curriculum or successful activities. New, new, odd conduct, unexpected or noisy responses can be original.

4. PROPOSED SYSTEM DESIGN

The Weka 3.7 open-source environment has prepared the proposed implementation, formally conducted, and intensive testing is displayed to expose mistakes and imperfections in the examiner suspension that necessitate being determined. Numerous validation experiments are conducted thoroughly in the Test Condition. Several experiments may be authorized to confirm documentation, guiding, probability projections, disaster retrieval, and demonstration relying on this system's appropriate circumstances. The Test Period ends with an examination to determine status to proceed to the application state.

4.1 Data collection:

We created some questionnaire surveys by using Google forms and some printing material. We collected almost 300+ samples for student evaluation data for analysis of proposed research.

4.2 Pre-Processing and Normalization

For this stage we will do data reduction and use techniques for example data balancing, data minimization and data cleaning to balance the data.

4.3 Extraction of feature and their selection:

In this progression we will use balanced and normalized data to extract the various feature from input data. We will use feature threshold to remove redundant as well as worst features for training the module.

4.4 Module Training:

In this process we will use machine learning classifier on extracted feature with the module will focus on supervised learning approach.

4.5 Testing of Module

For this step the individual instance process or may be entire test gets tested again to get the accuracy of the data. In this phase we will evaluate the efficacy of the system with the numerous datasets. This is to improve accuracy of module to do predictions near to actuals.

4.6 Prediction analysis:

Finally evaluate the arrangement accuracy of entire system using confusion metrics.

The dataset would be used to evaluate the proposed classification system. The qualities have considered based on standing like basic, intermediate and important etc. Below is the flow of the overall experiment explaining how the overall experiment will progress

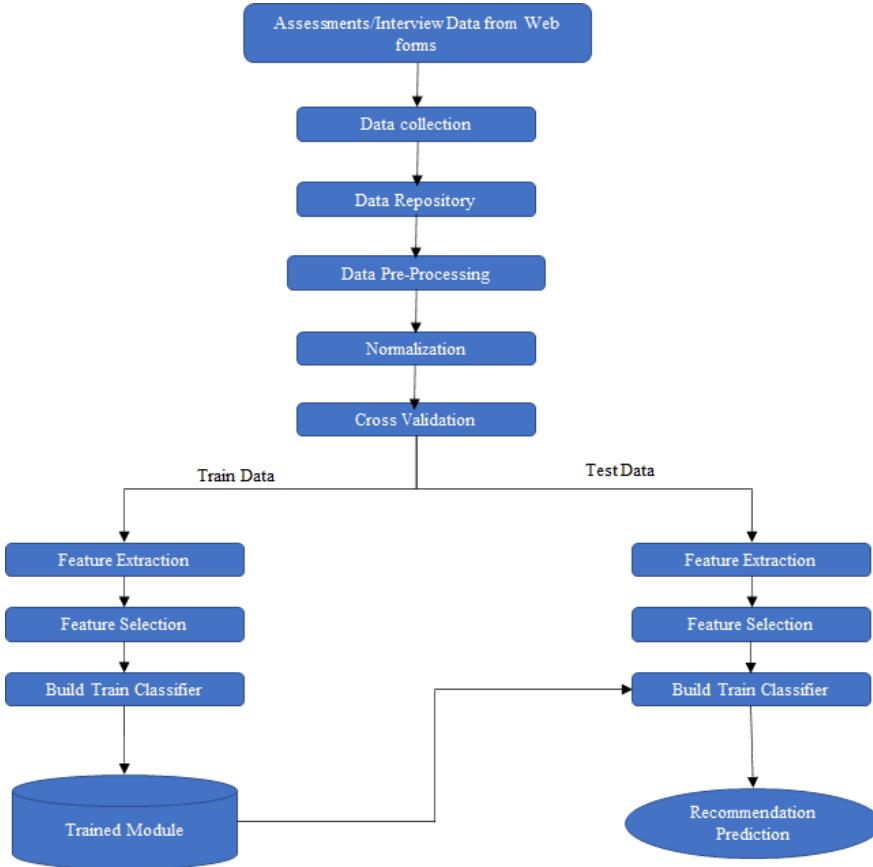


Figure 1: Experiment Design Flow

5. DESIGN OF EXPERIMENT

For the proposed implementation of the original few progressive distribution tasks, we became used Weka workbench environment. Investigations are handled in four sequential measures. In measure one, attribute evaluation is implemented utilizing the numerous managed distribution algorithms to interpret which attribute has the several meaningful implied influence on each class in the dataset.

Table 1: Existing methods evaluation of EDM

Algorithms Used	Accuracy
J48-NB-MLP-SMO	83.0%, 82.8%, 83.5%, 83.8%
MOOC	81.16%
KDD and RAD	80.2
C45, CART, NB	96.30%, 92.60%, 91.60%
Neighbourhood Cleaning Rule (NCL)	91.37%
NB, DT, KNN	73.61%, 81.94%, 80.56%
J48 - C4.5 - Random tree - and REPTree	44.10%
SVM – QDA – KNN - LR and RF	82.57%

J48 - Decision Tree - RT	85.60%
NB; BN; ID3; J48; NN	93%
NB – KNN – RF – NN – DT - Xmeans	83.65%
ID3 - C4.5 – KNN - NB	43.18
J48; REPTree; RT	62.30%
LR; SVM	76.67%
PNN – RF – DT – NB – TE - LR	89.15%
LR, NN, ANN, RNN, DNN	80.20%, 80.50%, 80.63%, 95.50%

Various methods and procedures of DM are appropriately designed in entire experimental analysis. The data set is tested and analysed using multiple data classifiers, which are J48 DT, MLP, NB, and ANN. A measurement of the correctness of all algorithms is conducted during the execution process. It is determined that utilizing the characteristic evaluation classification on the dataset maintains system realization. The essential attributes in the dataset are selected, then the suggested superior algorithms are operated on the dataset.

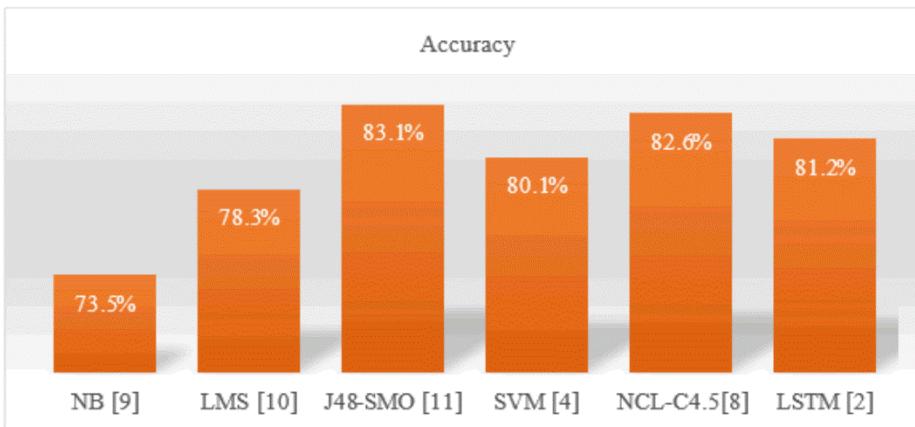


Figure 2: EDM classification accuracy of various machine learning algorithms

Different fascinating problem is perceived by the decisions, which show that the number of taught courses determines an evaluation performance. The above table 1 also designates that all sorting algorithms achieved lower prediction efficiency when operating on a dataset file than those algorithms' forecast accuracy of predicting dependant variable under study while run on numerous cross-validation. By examining of all classifiers, NB, SVM, and MLP algorithms implemented best amongst all classifiers with efficiencies of throughout 85% for all validation tests. Furthermore, we will do recognition, classification, analysis, forecasting and identification with the help of artificial intelligence. In the below Table we demonstrate some classification results.

Algorithm	Accuracy	Precision	Recall	F- Score
SVM	0.93	0.94	0.95	0.95
ANN	0.92	0.91	0.89	0.9

Table 2	Naïve Bayes	0.89	0.88	0.93	0.91
	Random forest	0.87	0.86	0.92	0.89
	J48	0.9	0.92	0.85	0.88

Accuracy classification of different Machine learning algorithm in Weka state

The present experiment highlights the numerous machine learning processes presentation on large students' data synthesized

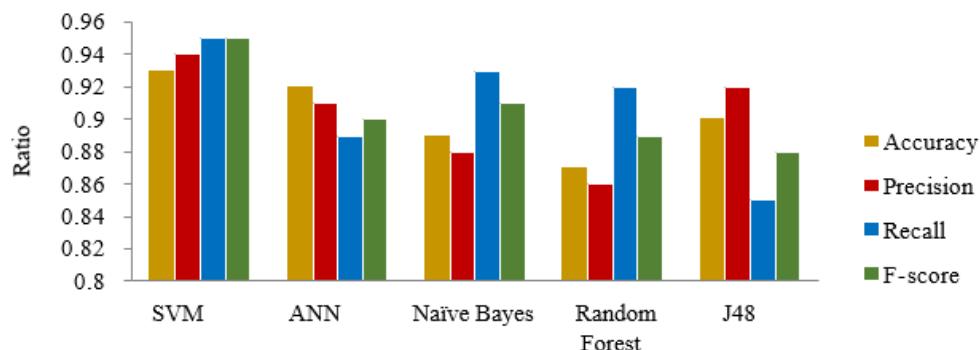


Figure 3: Results of data classification

Findings shows that Bayes and Artificial Neural Network (ANN) have reliable data set performance and predict robust prediction model. Authors argues that Machine learning application in context to reinforcement learning theory can augment the learning processes effectively. Reinforcement learning theory can solve many problems across the field be it manufacturing, supply chain, logistics, air control to swarm intelligence. Present study supports the reinforcement learning theory concepts and augmentation there of by applying machine learning application through data mining in education.

6. CONCLUSION

Educational data mining is an important computational tool for producing recommendations on educational practices. From the analysis, it can be concluded that both Bayes and Artificial Neural Network (ANN) have reliable data set performance. The ANN, furthermore, demonstrates greater consistency in results than in the Bayes theorem. In ANN, precision improved as the hidden layers improved. The study outcomes will help educational administrators recognize students at increased risk of not qualifying early for corrections measures. A comparative analysis of various architectures (NB, ANN, SVM, and J48) on data from participants to predict graduate higher education students was carried out throughout the research. The ANN and Bayes Classification are both appropriate architectures for data set processing and enhanced efficiency in RNN, DNN, etc. In the area of machine learning, researchers have applied various set of data for different purpose [13]. Starting from health sector to space the data mining tool is taking leap in the research world. Education being one of the prominent sectors impacting overall other sector and skill measurement aspect has greater advantage of applying machine learning applications [14]. Applying the data mining in this sector will be boom for predicting learning pattern [15] and related aspects of learning pattern across the disciplines. Future

study can be built based on the results produced by authors in this study. Empirical testing using statical technique would render insightful outcomes.

REFERENCES

1. Mukesh Kumar, A. J. Singh, "Performance Analysis of Students Using Machine Learning & Data Mining Approach," *Int. J. Eng. Adv. Technol* 8.3 (2019): 75-79.
2. Ling Cen, Dymitr Ruta, Lamees Mahmoud Mohd Said Al Qassem, Jason Ng, "Augmented Immersive Reality (AIR) for Improved Learning Performance: A Quantitative Evaluation," *IEEE Transactions on Learning Technologies*, April 2020, doi:10.1109/TLT.2019.2937525
3. Prosanta Kumar Chaki, Md Mozammel Hossain Sazal, Biman Barua, Md Shahadat Hossain, and Khaled Sohel Mohammad, "An Approach of Teachers' Quality Improvement by Analyzing Teaching Evaluations Data," In 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), pp. 1-5, IEEE, 2019.
4. Rastrollo-Guerrero, Juan L., Juan A. Gomez-Pulido, and Arturo Duran-Dominguez, "Analyzing and Predicting Students' Performance by Means of Machine Learning: A Review," *Applied Sciences* 10.3 (2020): 1042.
5. Maria Tsiakmaki, Georgios K Kotsopoulos, and Sotiris Kotsiantis, "Transfer Learning from Deep Neural Networks for Predicting Student Performance", *Applied Sciences*, 2020, 10.6: 2145, doi: 10.3390/app10062145
6. Sapna Arora, Manisha Agarwal, and Ruchi Kawatra, "Prediction of Educationist's Performance using Regression Model," 2020 7th International Conference on Computing for Sustainable Global Development (INDIACOM), IEEE, 2020, p. 88-93, doi: 10.23919/INDIACOM49435.2020.9083708
7. Ierin Babu, RV Siva Balan, and Paul P. Mathai, "Envisaging the Academic Performance of Student," 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), IEEE, 2019, p. 90-94, doi: 10.1109/ICCCIS48478.2019.8974531
8. Khafidurrohman Agustianto, Prawidya Destarianto, "Imbalance Data Handling using Neighborhood Cleaning Rule (NCL) Sampling Method for Precision Student Modeling," 2019 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE), IEEE, 2019.
9. Ajibade, Samuel-Soma M., Nor Bahiah Ahmad, and Siti Mariyam Shamsuddin, "A heuristic feature selection algorithm to evaluate academic performance of students," 2019 IEEE 10th Control and System Graduate Research Colloquium (ICSGRC), IEEE, 2019.
10. Al-Omar, Khalid, "Evaluating the Usability and Learnability of the Blackboard LMS Using SUS and Data Mining." 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), IEEE, 2018.
11. Ryan S. Baker, "Challenges for the future of educational data mining: The Baker learning analytics prizes," *JEDM Journal of Educational Data Mining* 11.1 (2019): 1-17.
12. Patil, J.M., Gupta, S.R.: Extracting Knowledge in Large Synthetic Datasets Using Educational Data Mining and Machine Learning Models. *Soft Computing for Intelligent Systems*. 167–175 (2021). https://doi.org/10.1007/978-981-16-1048-6_13
13. Sarker, I.H.: *Machine Learning: Algorithms, Real-World Applications and Research Directions*. SN Computer Science. 2, (2021). <https://doi.org/10.1007/s42979-021-00592-x>.
14. Yousafzai, B.K., Hayat, M., Afzal, S.: Application of machine learning and data mining in predicting the performance of intermediate and secondary education level student. *Education and Information Technologies*. 25, 4677–4697 (2020). <https://doi.org/10.1007/s10639-020-10189-1>
15. Zhang, Y., Yun, Y., An, R., Cui, J., Dai, H., Shang, X.: Educational Data Mining Techniques for Student Performance Prediction: Method Review and Comparison Analysis. *Frontiers in Psychology*. 12, (2021). <https://doi.org/10.3389/fpsyg.2021.698490>.

Traffic Surveillance and Vehicle Detection

YOLO and MobileNet-based ML pipeline transfer learning

^{*} Prof. Rakhi Bharadwaj¹, Aditya Thombre ², Umesh Patekar³, Yash Gaikwad⁴ and Sushil Suri⁵

¹Department of Computer Science,Vishwakarma Institute of Technology, Pune- 411037

^{1*} rakhi.bharadwaj@vit.edu

² Department of Computer Science,Vishwakarma Institute of Technology, Pune- 411037

² aditya.thombre20@vit.edu

³ Department of Computer Science,Vishwakarma Institute of Technology, Pune- 411037

³ umesh.patekar21@vit.edu

⁴ Department of Computer Science,Vishwakarma Institute of Technology, Pune- 411037

⁴ yash.gaikwad20@vit.edu

⁵ Department of Computer Science,Vishwakarma Institute of Technology, Pune- 411037

⁵ sushil.suri20@vit.edu

Abstract:In today's complex and interconnected transportation ecosystem, real-time vehicle sensing is critical for a complex and interconnected transportation ecosystem built on advanced technology networks of intelligent systems spanning a wide range of applications such as autonomous vehicles, traffic monitoring, and advanced driver assistance systems. In this paper, we use machine learning approaches to create a pipeline for vehicle identification and classification.count the number of cars in a frame and divide them into two categories: SUVs and sedans. This article requires knowledge of machine learning fundamentals, deep learning, convolutional networks, and transfer learning.in this paper broken down the pipeline in order to build and implement a computer vision pipeline.

Keywords:MobileNet,Traffic Surveillance, Transfer Learning, Traffic Surveillance, Vehicle

Detection

1 Introduction

In recent years, video cameras are often used in traffic surveillance systems as they are an important source of traffic flow data. Rapid advances in computer vision, computing, and camera technologies, as well as advances in automated video analysis and processing, have increased interest in video-based traffic surveillance applications. For intelligent transportation systems, the use of computer vision technology for traffic monitoring is becoming increasingly important. For incident detection, behavior analysis, and comprehension, these systems leverage visual appearance in vehicle identification, recognition, and tracking. Additionally, it offers characteristics for traffic flow, such as vehicle type, count, trajectory, etc. Even though there has been a lot of research done to enhance video-based traffic surveillance systems, there are still several problems with actual applications. In a recent assessment, the state-of-the-art networked and hierarchical surveillance architecture for vehicles was described, along with a thorough examination of unique computer vision difficulties. contains a poll on vehicle recognition, tracking, and on-road behavior analysis. A review of computer vision methods focused on infrastructure-side urban traffic analysis. The important concepts of computer vision and pattern recognition have been addressed, along with a thorough explanation of the technical difficulties and a comparison of the available solutions. This study provides a thorough analysis of several video-based traffic monitoring approaches from a computer vision perspective. It includes several methods for detecting, identifying, and tracking vehicles. Improvements and alterations are also discussed, along with the benefits and drawbacks.

2 Literature Review

A. This paper presents a study on a multi-camera system to block a specific section of the road in order to analyze the traffic situation in 3 sections. The second section is a detailed review of the literature on multichamber systems. The third section here is the proposed system using a two-chamber experimental setup with adjustments. Deep neural networks are used in traffic behavior analysis experiments. The focus of this paper is the physical design, calibration, and advantages/disadvantages of multi-chamber systems. In conclusion, future developments and improvements in the field of traffic analysis using multi-camera systems are discussed.[1]

B. In this paper, contemporary visual tracking emerges from the traditional mathematical approaches of neural networks as an active automated research area in computer vision. In this study, a new modified neural network method is

presented for object detection and classification of input images and videos from multiple cameras with overlapping regions of interest. The modified neural network approach provides a multilayer architecture as input, preprocessing, and manipulation layers to simplify the processing required to prepare the neural networks for training. This strategy uses predefined tasks to delegate tasks to shifts, simplifying training, reducing computational requirements, and helping to deliver performance. Its two neural network modules process the input. The first module is a modified neural network, which differs from traditional neural networks in the connections between neurons and their tasks. It's still a neural network that splits the data and shares thresholds to show the difference. This means that there are markers between the two inputs and the simplified training. The second module is a conventional recognition and classification neural network that tracks recognized objects. In this paper, we propose a system that provides composite images from the outputs of multiple cameras using unconventional mathematical and algorithmic approaches..[2]

C.This article introduces a system-wide reliable mechanism for real-time vehicle detection. This approach combines the MOG2 background subtraction model (Gaussian blend) with a modified SqueezeNet (H-SqueezeNet) model. Generate scale-independent regions of interest (RoIs) using the MOG2 model from video frames. H-SqueezeNet is provided for reliable vehicle classification identification. The effectiveness of this method was validated using the CDnet2014 dataset, the UA-DETRAC dataset and video footage from Suzhou Intersection. Test results show that this approach can provide a high detection accuracy and average detection rate of 39.1 frames per second in a traffic monitoring system.[3]

D. This document refers to license plates in image or video collections as ALPR or Automatic License Plate Recognition (ALPR) (ANPR). ANPR technology uses intelligent transportation systems to minimize the need for human contact. The purpose of this research is to find an optimal license plate recognition algorithm. In this study, we use four deep neural networks, including CNN, VGG16, VGG19, and YOLOV3, to detect license plates, evaluate model accuracy, and select the best model. This article refers to license plates in an image or video collection as ALPR or Automatic License Plate Recognition (ALPR) (ANPR). ANPR technology enables intelligent transportation systems and minimizes the need for human contact. The goal of this research is to discover an optimal license plate recognition algorithm. This study uses four deep neural networks, including CNN, VGG16, VGG19, and YOLOV3, to recognize license plates, evaluate model accuracy, and select the best model.[4]

E.In this work, FCN was used to extract lane marking features. Individual classification experiments can be performed on each pixel of the input image. After removing the original complete convolutional layer, the loaded feature maps were

immediately restored. The Tusimple dataset was used to train the network parameters. Use the Hough transform to specify the tuning interval and the smallest track tuning inside the tuning interval to accomplish multi-band tuning by combining it with the least squares approach. There is now a net. The experimental findings demonstrate that FCN is superior to other neural networks in terms of lane edge pixel extraction, and that its lane identification rate is superior to that of other techniques.[5]

F. Video is particularly important in this context. In recent years, it has been extensively employed in traffic monitoring and control systems. A significant amount of research is being undertaken on traffic control systems. Examples of these applications where video is processed include early warning or information extraction in real-time, as well as analytics through vehicle identification and categorization. Vehicle detection and categorization are discussed in this article.[6].

G. This paper presented a vehicle detection algorithm that can identify automobiles as well as traffic offenses. Most computers and AI (artificial intelligence) vision systems require the capacity to recognise objects. The acquired results are examined and tabulated. It has a 91% accuracy on the NIR picture and an 88% accuracy on the blurred image dataset. The detection time is reduced by seconds for high-density traffic flows. As a result, system operating speed is affected by traffic density. It may also be used on bigger datasets by training on GPUs and high-end FPGA packages..[7]

H. This report outlines a vehicle detection algorithm that can identify automobiles as well as traffic offenses. Most computers and AI (artificial intelligence) vision systems require the capacity to recognise objects. The acquired results are examined and tabulated. It has a 91% accuracy on the NIR picture and an 88% accuracy on the blurred image dataset. The detection time is reduced by seconds for high-density traffic flows. As a result, system operating speed is affected by traffic density. It may also be used on bigger datasets by training on GPUs and high-end FPGA packages.[8]

3 Methodology

To implement and design a computer vision pipeline. perform these tasks:

1. Reading Video:

To read video frames we used Opencv and the original video has been shown, then a queue of the frames passed to the next task.

2. Detecting Object:

TinyYOLO is used for object detection. TinyYOLO is a pre-trained model on the COCO dataset. YOLO's job is to recognize the car outside its frame.

3. Classifying Car Types:

Once we get cars using the bounding box for each frame, we have to classify them as SUVs or sedans. For performing this task, MobileNetV2 classifier is used and applies the concept of transfer learning to modify the model according to the functionality.

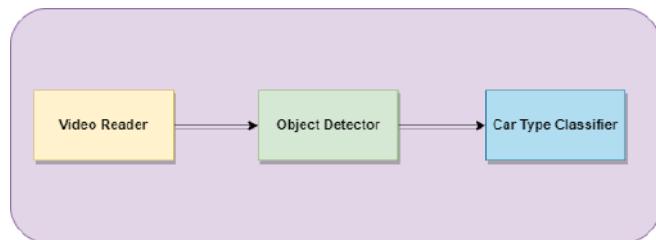


Figure.1 Computer vision pipeline

3.1 Pipeline Design, Model Configurations and Working

- OpenCV was used for reading the video from the video frames with the rate of 30 FPS.
- preparation of dataset: images of the two classes are included in the dataset, these classes are SUV and Sedan. there are total 1748 images of sedan and 1432 images of SUV in the dataset. images are collected from the Stanford car dataset and the Google images web scraping. it is included in the data preparation.
- TinyYOLO is preferred over the YOLO for object detection in the video.. Because it is faster and lighter than the latter. TinyYOLO can handle 220 FPS, while YOLO handles 20-40 FPS. Since we need real-time detection.
- After reading all frames in OpenCV, the queue of frames is passed to the detection function. For detecting classes in a given frame, The detection function uses YOLO's detect_image function.
- In the end, the location of detected cars in particular frames are passed to classification function, to classify the cars into specified classes. Model trained using MobileNet model was used in this function.

3.3 Transfer Learning and MobileNet Model:

- In figure 2.basic structure of MobileNet model is shown and reused this model using transfer learning concept.

Type	Filters	Size	Output
Convolutional	32	3×3	256×256
Convolutional	64	$3 \times 3 / 2$	128×128
1x			
Convolutional	32	1×1	
Convolutional	64	3×3	
Residual			128×128
Convolutional	128	$3 \times 3 / 2$	64×64
2x			
Convolutional	64	1×1	
Convolutional	128	3×3	
Residual			64×64
Convolutional	256	$3 \times 3 / 2$	32×32
8x			
Convolutional	128	1×1	
Convolutional	256	3×3	
Residual			32×32
Convolutional	512	$3 \times 3 / 2$	16×16
8x			
Convolutional	256	1×1	
Convolutional	512	3×3	
Residual			16×16
Convolutional	1024	$3 \times 3 / 2$	8×8
4x			
Convolutional	512	1×1	
Convolutional	1024	3×3	
Residual			8×8
Avgpool		Global	
Connected		1000	
Softmax			

Figure 2. Structure of original MobileNetV2 Model

- The MobileNet model was used as a starting point, we added our own layers and discarded the last layer which is shown in figure 3. In this model a single output layer is present with 2 additional hidden layers of 512 neurons and 1 hidden layer with 256 neurons.finally single output layer was added. due to its binary classification problem.

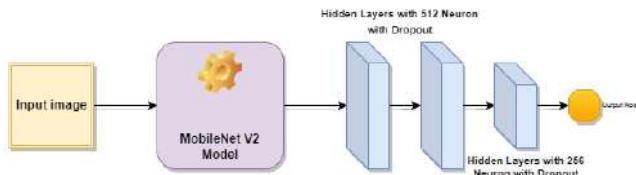


Figure 3. Structure of modified model

3.4 Model Tuning and Hyperparameters

A. Dropout: a new layer dropout using A regularization method with probability 0.2 was used. This helped reduce model overfitting that could be investigated due to the gaps between validation and training error and validation accuracy.

B. Data Augmentation: An attempt was made to use data extensions to perform different operations such as zooming, panning, etc., but this operation was discarded as the results were not satisfactory. Machine learning applications, especially deep learning applications, continue to diversify and grow rapidly. Data-centric approaches to model development, such as data augmentation techniques, can be excellent tools for the challenges facing the world of artificial intelligence. Data augmentation helps improve the performance and results of machine learning models by forming new and different examples for training datasets. When the dataset is included in a machine learning model.

C. Epochs: 20 epochs were set. This is because a large number of epochs may lead to unsatisfactory results in terms of poor accuracy improvement, ground truth and overfitting.

D. L1/L2 Regularization: we tried adjusting weights for L1 and L2, with very large validation errors and poor accuracy. We got bad results including poor accuracy and large validation errors.

E. Model compilation: As a progress metric, binary cross-entropy with accuracy used in the loss function. Finally, with the correct name model weights are saved.

F. Gradient descent optimization algorithms: Two optimization algorithms were tried, Adam optimizer and RMSProp. For our project, Adam gave better results than RMSProp, so we used it. 0.0001 was the learning rate. because a low learning rate requires a higher epoch and slows down learning.

3.5 Number of the epoch vs Accuracy and Cross-Entropy

In figure 4 below,The time/epoch for Adam and RMSProp optimizations versus the variation of accuracy and cross-entropy is shown.Adam was preferred over RMSProp due to, difference between validation accuracy and training accuracy is smaller than RMSProp, resulting in less overfitting models. As expected, it outperforms F1 in the number of sedans and SUVs, and also proves to be a better model overall.

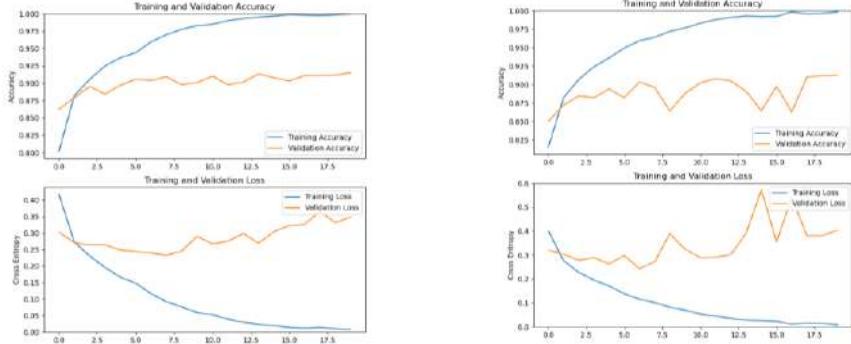


Figure 4. wrt to time/ epoch, variation of accuracy and cross-entropy (left) for Adam (Right) and RMSProp(Lower)

4 Result and discussions

4.1 Deep learning models final accuracy

TABLE I. Adam optimizer ($lr=0.0001$)

Training loss =0.0075	Training accuracy=0.9994
Validation loss =0.3485	Validation accuracy =0.9148

TABLE II. RMSProp optimizer ($lr=0.0001$)

Training loss =0.0077	Training accuracy =0.9981
Validation loss =0.4031	Validation precision =0.9125

4.2 Pipeline Optimization

- For Intersection over Union Threshold (IOU) and the Non-Maximal Suppression (NMS) parameters of the YOLO model for better results various experiments were performed. The threshold for object selection removes all boxes with low probabilities and intersection probabilities are modified by this parameter. In the TinyYOLO, an IOU of 0.2 and NMS score of 0.2 gives better result, which improves the F1 score to ground truth.

- Experiment with different hyperparameters for our deep learning model, including epochs, optimizers, regularization, dropout, and data augmentation focused on providing a well-defined and accurate input dataset to our training model.
- We have incorporated a fast processing approach that uses a thread pool (executor framework) to process frames in parallel, reducing execution time by a factor of 1. It will be more useful in the future when you need to process longer videos. Even so, it currently takes about 3 minutes to process (recognize and classify) a 900-frame video, so the pipeline needs to be sped up a bit more.

Training Loss = 0.0075, Training Accuracy = 0.9994, Validation Loss = 0.3485, Validation Accuracy= 0.9148, Training Loss = 0.0077, Training Accuracy = 0.9981, Validation Loss = 0.4031 and Validation Accuracy = 0.9125

- Early, hair cascades were used to detect cars but did not get better results, so YOLO was used for object detection because it gives much better results than the first method.

4.3 Output of Pipeline

In the figure5, the video displayed to the user and output contains detected cars and their types in the pipeline. output of the system also stored in the excel sheet.includes tables like comparing Sedan, SUV and total F1 score with number of cars per frame.which is shown in the figure 6.



Figure 5. Output with the number of cars with their types

```
*****
Accuracy/F1 Score with respect to Ground Truth
*****
Accuracy : 0.5433333333333333
F1 Score for Total Cars in each frame: 0.5601001854715687
F1 Score for SUV in each frame: 0.3164389897441949
F1 Score for Sedan in each frame: 0.47096479319590045
```

Figure 6.The F1 Score for Total Cars

4.4 Execution Time

This system processed 900 frames in 460 seconds. Comparing the time taken by the ML model to classify objects and time taken by YOLO to process the object detection is shown in figure 6.

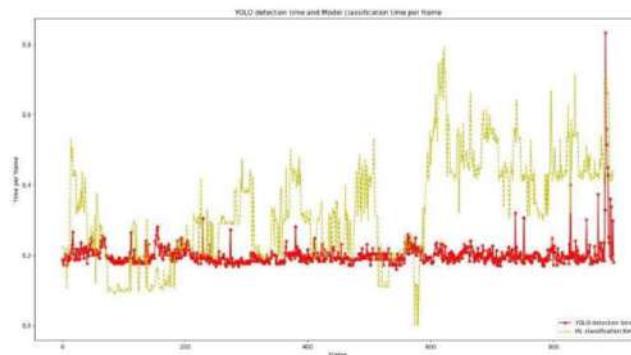


Figure 7 . Detection and Classification time for each frame

5 Conclusion and Future Scope

Although it is a computer vision pipeline, the implementation of the code is in object-oriented based approach and has separate classes for video reading, object identification, automatic classification, etc., allowing for programming reuse. and contributing to the extension of this project.

In security systems such as traffic monitoring, the goal is to get the desired information quickly. Our goal is to push the limited capabilities of computer systems to higher levels using distributed systems and parallel operations. Additionally, it can be implemented on larger datasets by training on GPUs and high-end FPGA kits.

Acknowledgment

It would be our utmost pleasure to express our sincere thanks to our guide Prof.Rakhi Bharadwaj who gave us the opportunity to do more research on the topic “Traffic Surveillance and Vehicle Detection ML pipeline using YOLO and MobileNet transfer learning”, which also helped us in doing plenty of ideas and that we came to understand about such a lot of new things..

References

1. Bhardwaj, Rakhi Joshi, and D.S. Rao. "Deep Learning-Based Traffic Behavior Analysis under Multiple Camera Environment." *International Journal of Next-Generation Computing*, October 31, 2022. <https://doi.org/10.47164/ijngc.v13i3.719>.
2. Bhardwaj, Rakhi Joshi, and D.S. Rao. "Modified Neural Network-Based Object Classification in Video Surveillance System." *International Journal of Next-Generation Computing*, October 31, 2022. <https://doi.org/10.47164/ijngc.v13i3.890>.
3. Wang, Zhiyuan, Jifeng Huang, Neal N. Xiong, Xiaoping Zhou, Xiao Lin, and Theodore Lee Ward. "A Robust Vehicle Detection Scheme for Intelligent Traffic Surveillance Systems in Smart Cities." *IEEE Access* 8 (2020): 139299–312. <https://doi.org/10.1109/access.2020.3012995>.
4. G and M, "A Comparative Study of Four Deep Neural Networks for Automatic License Number Plate Recognition System – IJERT."
5. Chao, Fan, Song Yu-Pei, and Jiao Ya-Jie. "Multi-lane detection based on deep convolutional neural network." *IEEE access* 7 (2019): 150833-150841.
6. Kul, Seda, Süleyman Eken, and Ahmet Sayar. "A concise review on vehicle detection and classification." In 2017 International Conference on Engineering and Technology (ICET), pp. 1-4. IEEE, 2017.
7. Kumar, Chethan, and R. Punitha. "Performance Analysis of Object Detection Algorithm for Intelligent Traffic Surveillance System." In 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), pp. 573-579. IEEE, 2020.
8. K. -J. Kim, P. -K. Kim, Y. -S. Chung and D. -H. Choi, "Multi-Scale Detector for Accurate Vehicle Detection in Traffic Surveillance Data," in *IEEE Access*, vol. 7, pp. 78311-78319, 2019.
9. Rodríguez-Rangel H, Morales-Rosales LA, Imperial-Rojo R, Roman-Garay MA, Peralta-Peñañuri GE, Lobato-Báez M. Analysis of Statistical and Artificial Intelligence Algorithms for Real-Time Speed Estimation Based on Vehicle Detection with YOLO. *Applied Sciences*. 2022; 12(6):2907.
10. Appathurai, Ahilan, Revathi Sundarasekar, C. Raja, E. John Alex, C. Anna Palagan, and A. Nithya. "An Efficient Optimal Neural Network-Based Moving Vehicle Detection in Traffic Video Surveillance System." *Circuits, Systems, and Signal Processing* 39, no. 2 (August 13, 2019): 734–56. <https://doi.org/10.1007/s00034-019-01224-9>.
11. A. Nithya. "An efficient optimal neural network-based moving vehicle detection in traffic video surveillance system." *Circuits, Systems, and Signal Processing* 39, no. 2 (2020): 734-756.

12. Mohamed, Atibi, Atouf Issam, Boussaa Mohamed, and Bennis Abdellatif. "Real-time detection of vehicles using the haar-like features and artificial neuron networks." *Procedia Computer Science* 73 (2015): 24-31.
13. Azimjon, Jahongir, and Ahmet Özmen. "A real-time vehicle detection and novel vehicle tracking systems for estimating and monitoring traffic flow on highways." *Advanced Engineering Informatics* 50 (2021): 101393.
14. Billones, Robert Kerwin C., Argel A. Bandala, Edwin Sybingco, Laurence A. Gan Lim, Alexis D. Fillone, and Elmer P. Dadios. "Vehicle detection and tracking using corner feature points and artificial neural networks for a vision-based contactless apprehension system." In the 2017 *Computing Conference*, pp. 688-691. IEEE, 2017.
15. Sudha, D., and J. Priyadarshini. "An intelligent multiple vehicle detection and tracking using modified vibe algorithm and deep learning algorithm." *Soft Computing* 24, no. 22 (2020): 17417-17429.
16. Lin, Cheng-Jian, Shiou-Yun Jeng, and Hong-Wei Lioa. "A real-time vehicle counting, speed estimation, and classification system based on virtual detection zone and YOLO." *Mathematical Problems in Engineering* 2021 (2021).
17. Ma, Yongchang, Mashrur Chowdhury, Adel Sadek, and Mansoureh Jeihani. "Real-time highway traffic condition assessment framework using vehicle-infrastructure integration (VII) with artificial intelligence (AI)." *IEEE Transactions on Intelligent Transportation Systems* 10, no. 4 (2009): 615-627.
18. Min, Jin Hong, Seung Woo Ham, Dong-Kyu Kim, and Eun Hak Lee. "Deep Multimodal Learning for Traffic Speed Estimation Combining Dedicated Short-Range Communication and Vehicle Detection System Data." *Transportation Research Record* (2022): 03611981221130026.
19. Alsanabani, Ala A., Salah A. Saeed, Moeen Al-Mkhlaifi, and Mohammed Albishari. "A Low Cost and Real Time Vehicle Detection Using Enhanced YOLOv4-Tiny." In *2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 372-377. IEEE, 2021.
20. Goerick, Christian, Detlev Noll, and Martin Werner. "Artificial neural networks in real-time car detection and tracking applications." *Pattern Recognition Letters* 17, no. 4 (1996): 335-343.
21. Zuraimi, Muhammad Azhad Bin, and Fadhlwan Hafizhelmi Kamaru Zaman. "Vehicle detection and tracking using YOLO and DeepSORT." In *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, pp. 23-29. IEEE, 2021.
22. Smitha, J. A. and N. Rajkumar. "Optimal free forward neural network based automatic moving vehicle detection system." *Multimedia Tools and Applications* 79, no. 25 (2020): 18591-18610.
23. Kejriwal, Rahul, H. J. Ritika and Arpit Arora. "Artificial Intelligence (AI) Enabled Vehicle detection and Counting Using Deep Learning." In *2022 International Conference on Computer Communication and informatics (ICCCI)*, pp. 1-5. IEEE, 2022.

Perceptors: A Real Time Object Detection System with Voice Feedback and Distance Approximation for Blind

*Prof. Rakhi Bharadwaj¹, *Harshal Sonawane², *Manasi Patil³, *Shashank Patil⁴, and *Vedant Jadhav⁵

¹*Department of Computer Engineering, Vishwakarma Institute of Technology Pune, India*
rakhi.bharadwaj@vit.edu

²*Department of Computer Engineering, Vishwakarma Institute of Technology Pune, India*
harshalsonawane127@gmail.com

³*Department of Computer Engineering, Vishwakarma Institute of Technology Pune, India*
manasi.patil21@vit.edu

⁴*Department of Computer Engineering, Vishwakarma Institute of Technology Pune, India*
shashank.patil21@vit.edu

⁵*Department of Computer Engineering, Vishwakarma Institute of Technology Pune, India*
vedant.jadhav20@vit.edu

Abstract: Engineering solutions have impacted everyone's lives, but those designed to aid people with disabilities have been especially significant. However, the current pricing of modern assistive gadgets does not match the market's needs. In its latest study, WHO puts the number of visually impaired people at 285 million. Of these, 246 million are blind and an estimated 39 million are visually impaired. Blind people face many problems and difficulties in business and social activities. It also has the potential to contribute to society if the opportunity arises. In this paper we propose to you "Perceptors" - An A.I. powered smart glasses that can detect objects in real time and alert users when objects are nearby. This research is an attempt to give visually impaired people a chance to experience a moderately normal life by warning users in advance of approaching objects and helps users identify objects in real time.

Keywords: Raspberry Pi, Python, Object Detection, Voice Processing, Distance Approximation.

1. Introduction

For the past few decades, visual impairments have been one of the most common issues. Devices used by the blind to navigate are not sufficiently accessible since they primarily rely on infrastructural needs. The most widely used is a simple stick or cane. Blind people swing the stick back and forth to use it to perceive obstacles,

but unfortunately blind people may notice obstacles too late. With this, recent technological advances have changed the normal cane to a stick which is attached with an ultrasonic sensor for object detection and distance proximation. Then too there are some restrictions.

Visually impaired people may find it difficult to move through rooms or different types of roads without their vision. Most shocking thing is that 75% of these blind people have preventable eyesight. There are not enough opticians and very few people take an initiative and sign a contract for eye donation after their death in order to treat corneal sight. The main goal of " Perceptrors " is to help blind people to move like normal people. This design is based on a preliminary study of visually impaired individuals. This research uses camera-based glasses, a methodology that people can use while walking. It has Object Detection technology that recognizes nearby objects. When an object is recognized, it converts text into speech, which can be heard by blind people. The purpose of "Perceptrors" is to assist these people in various areas of life. For example, these glasses are effective for pedestrians. Helps pedestrians recognize approaching objects. Ultrasonic sensors can also be used to calculate distances and send alerts when objects are approaching. With the help of this system, they can walk alone through streets, traffic areas, or parks without depending on anyone. Because you are interested, you can enjoy your life with the help of this system. This wearable device is designed to assist the visually impaired in their mobility. Most commercial devices lack some necessary features, but these devices conveniently overcome them. The device also warns the blind if there are obstacles and guides the blind through streets and different areas via voice support.

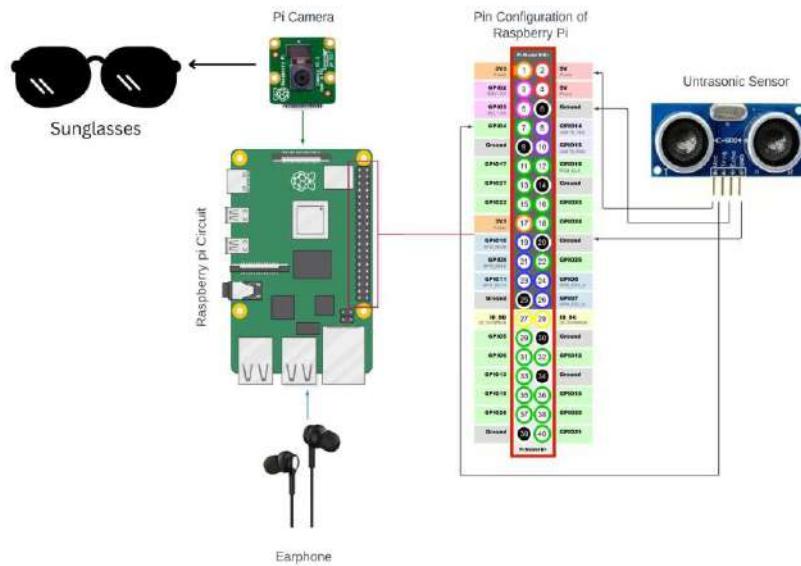


Fig. 1: Experimental Setup

2. Literature Review

In a rapidly growing country like ours, multiple attempts have been made with the welfare in mind of specially abled persons in surroundings. Operation "Project Prakash" is one of these endeavours.[2], which aims to help blind children by gaining knowledge of a set of obstacles around them by using their intelligence. Sheth et al [3] worked on finding various ways on how a blind person may be used to find any kind of pits, potholes and several other things by using a smart stick where they have used ultrasonic sensors for object detection. This gadget has a multinational audio system. feedback cannot be used because it has a limitation of recording only 680 seconds. The research in [4] has an a pits sensor, an ultrasonic sensor, as well has a water sensor. As well has a GPS system, but this requires the user to enter their location themselves. Instructions for doing so are not mentioned here. In [5] we see that the frame itself consists of a video camera, a CPU with appropriate size, and software that takes pictures of nearby objects with transparency. eyepiece. The main limitation of this device is that it is completely unsuitable for visionless people. This is only recommended for blind people or a person with night blindness. There is another upcoming attempt to assist the visually

impaired called Haptic Assisted Location of Hurdles, or H.A.L.O. [6]. It includes a rangefinder that receives data from an ultrasonic sensor. sensor and outputs her 's feedback to a pulse vibration motor placed on a visionless person's head.

Sr No.	Title	Findings	Year Of Publication
1	Smart Glasses for Blind People [8]	In this paper, authors discussed a device that can detect an object and convert an image to speech.	2021
2	Real Distance Measurement Using Object Detection of Artificial Intelligence [9]	The idea behind this paper is to measure the distance to an object.	2021
3	IoT Enabled Automated Object Recognition for the Visually Impaired [11]	To recognise objects, this system employs Tensorflow-Single lite's Shot Detector (SSD) architecture with MobileNet.	2021
4	Custom object detection in browser using TensorFlow.js (BLOG) [13]	This blog post demonstrates how to train a bespoke object-detection model using TensorFlow to create an end-to-end approach.	2021
5	Object Detection for Blind Users [15]	This study describes a technology that would enable blind persons to sight using object-like pictures and video scenes.	2020
6	A review on methods for speech-to-text and text-to-speech conversion [17]	Through this paper authors aim to study the different methodology for Text-To-Speech Conversion.	2020
7	Distance Measurement using Sensors and Arduino [18]	The primary goal of this research was to create a distance system of measurement that interconnected with an Arduino and used ultrasonic waves..	2017
8	Design and Implementation of Text To Speech Conversion for Visually Impaired. [19]	Text-To-Speech usingNLP, DSP and reads out to the user.	2014

9	Implementation of Text to Speech Conversion [20]	In this study, a text-to-speech conversion system that can get the text through image using Optical Character Recognition (OCR) in MATLAB	2014
---	--	---	------

3. Problem Statement

Having a visual impairment limits the way you interact with others, access information, and develop your own knowledge and experiences, so you can multitask to cope with a variety of situations. The need for tools is a key issue. The tools and technologies currently on the market are expensive for the average or low-income people who make up the majority of users, so new, cheaper devices with similar tasks are needed. The existing systems failed to detect objects in real time. Few systems used the object detection models to detect static objects from images. There was no system present that could sense the distance between the object and the concerned person. Existing systems failed to achieve fast computation and failed to present accurate results. Cost of the system is a major concern; Existing systems are costly as compared to the proposed system.

Smart Cane: Uses an ultrasonic sensor to detect any obstacles above chest level and warns the user via a vibrating handle. No object detection capabilities.

InnoMake shoes: Includes a pair of ultrasonic sensors at the tip of each shoe that vibrates and makes noises to warn the individual of the obstacle in front of them. Costs around €3,200 which is very expensive to average people.

OrCam MyEye Pro: Glasses that capture image when user clicks button and then communicate the info audibly through a tiny speaker that rests above the ear. No real time object detection, an image is clicked first and then objects are detected thus it increases the response time and computation overhead.

The main goal of the proposed system is to offer a dependable, economical, and low-power solution that allows visually impaired people to walk normally as a pedestrian would. A sizable section of society can afford the price of such a system, providing a disposable device and ensuring excellent guidance.

4. Proposed Methodology

The functioning of proposed system can be understood by these four main sub module:

- A. Object Detection
- B. Object Recognition
- C. Distance Approximation
- D. Voice Processing (Text-To-Audio)

A. Object Detection:

The previous object detection techniques rely on palm functions and flat, trainable systems. By building sophisticated ensembles that blend numerous poor visual features of high context via features extracted and scenario classifications, their performances could be simply drowned out. As machine learning quickly advances, more potent methods for learning meanings and richer, more flexible functions are defined to tackle problems with old architectures. For the purpose of object detection the proposed system uses Tensor flow Lite Open Source Library. TensorFlow Lite is used to help TensorFlow models convert them to a much more compact deep learning (ML) model structure. Additionally, you can alter which was before models that are already in TensorFlow Lite or build your own TensorFlow models and then export it in TensorFlow Lite form. Images, videos, audio, and texts may all be used as inputs in TensorFlow models to perform tasks like object recognition, language processing, pattern matching, and more. Proposed system implements the tensorflow-lite model for the purpose of object detection. Tensor flow lite is a lightweight model specially designed for embedded applications having less computation power. Along with the tensorflow lite, the Single Shot Detector (SSD) model with MobileNet is used.

The proposed system can also detect objects from pictures provided that the images are of good quality. The proposed system marks the area of interest with a rectangular grid and then the SSD MobilNet model recognizes the object present in the image using an array of detection summary info, name - detection_out, shape - 1, 1, 100, 7 in the format 1, 1, N, 7, where

N is the number of detected bounding boxes.

For each detection, the description has the format: [image_id, label, conf, x_min, y_min, x_max, y_max].

B. Object Recognition:

Various methods could be used for object identification. Computational modeling and machine learning methods have recently gained popularity as solutions to object identification issues. Both methods teach users how to recognise items in pictures,

but they operate in different ways. One of the quickest and most dependable versions of MobileNets is mobilenet-v3-large-1.0-224-tf (MobileNets V3). It is built on a creative architectural design and a blend of several search strategies. The increased resource application scenarios are satisfied by mobilenet-v3-large-1.0-224-tf.

Image, name: input , shape: [1x3x224x224], format: [BxCxHxW], where:

- B - batch size
- C - number of channels
- H - image height
- W - image width

Faster R-CNN 7 FPS with mAP 73.2% or YOLO 45 FPS with mAP 63.4% were both slower than the SSD method's 59 FPS with mAP 74.3% on the VOC2007 test. It aggregates detections at various sizes by using several image features from of the later stages of the network and small convolution filters to forecast item categories and thresholded positions for various aspect ratios. The suggested model can identify roughly 10–15 frames per second. There is room for advancement by using a cutting-edge microprocessor. (Pi4) Raspberry.

C. Distance Approximation

Manual distance measurement always comes at the expense of human error. An ultrasonic sensor contains a transducer that, when the transducer element vibrates, emits inaudible, high-frequency sound waves in one direction. When a wave hits an object and bounces off it, the transducer receives an echo signal. The sensor then uses the time between the first sound burst and the return of the echo to determine the distance to the object. Various sensors were considered for Distance approximation, out of these sensors, ultrasonic sensors showed promising results while barring minimum cost. In our research, we have used ultrasonic sensors for distance approximation. We chose Ultrasonic sensor instead of IR sensor, because in case of transparent objects, IR sensor lacks in reflecting the waves backs. Hence, Ultrasonic sensors are better and more reliable than any of the contemporary .Proposed system implements the tensorflow-lite model for the purpose of object detection. Tensor flow lite is a lightweight model specially designed for embedded applications having less computation power. Along with the tensorflow lite, the Single Shot Detector (SSD) model with MobileNet is used.

The average speed of sound is around 340 m / sec, or 29.412 milliseconds per centimeter. The formula Distance = (time x speed of sound) / 2 is used to detect the length that sound travels. Since sound must travel equally backward and forward, the number "2" is included in the equation. The sound leaves the sensors first, then leaves the surface and recovers. The equation centimeters = ((microseconds / 2) / 29) can be used to quickly read the distance in centimeters. For instance, if an ultrasonic sound bounces after 100 s (microseconds), the range is ((100 / 2) / 29)

centimeters, or approximately 1.7 cm.

$$Distance (d) = (Speed * Time)/2$$

Example:

$$Distance = (34 \text{ ms} * 1.5 \text{ ms})/2$$

$d = 25.2 \text{ cm}$, where 1.5ms = receiver pulse duration, 34 m/s is speed of sound

D. Voice Processing Method

gTTS is an API to convert text in audio in Python. Several languages, including English, Hindi, Tamil, French, German, etc., are supported by the gTTS Apis. The speech is subsequently presented in one of multiple audio rates of speed or slow—that are offered. gTTS has some of the best features like being customizable, can read unlimited length of text, all while keeping proper intonation, abbreviations, decimals and more and can also provide pronunciation corrections wherever required. These features make it the best choice for our research.

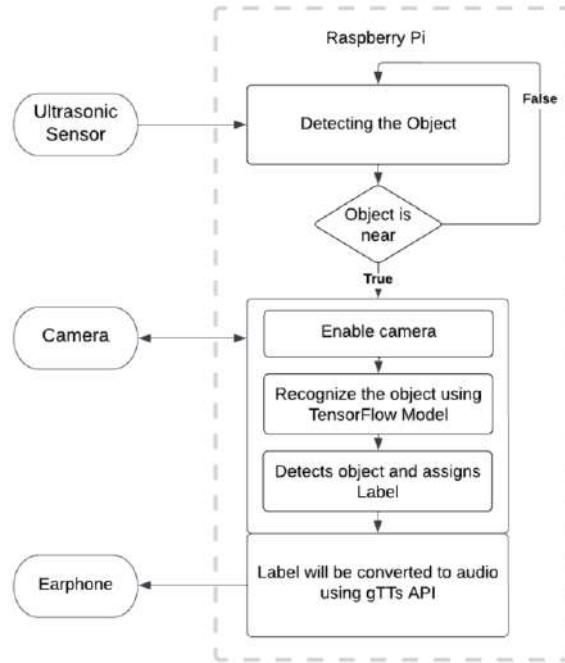


Fig 2: Architecture of Proposed System

As mentioned in Figure 2, shows the overall execution and working of the proposed system. It shows the various items used such as headphones, Raspberry Pi circuits,

and sensors. It is based on the TF-Lite Model as well as SSD Mobnet Object detection.

The proposed system would be equipped with Ultrasonic sensors deployed that would continuously sense the distance between the subject and the object if any object is in front of the subject. If the object is within the defined threshold (150 centi-meter) then the system would alert the Camera module to start detecting objects using the object detection model. The camera model would then detect and recognize the approaching object along with the distance at which it is approaching. After sensing the object, the camera module would generate a label. The generated label would have details such as the identified object, confidence rate of the detected object, the distance between the subject and object. The system would then send the label for voice simulation. The voice simulation module in turn would convert the labels into a speech which could be heard by the subject thus alerting the user of any approaching object by identifying it.

5. Technological Requirements

1. Hardware Requirement

A. *Raspberry Pi 3 Model B+*

The Raspberry Pi, a simple Mastercard measurement PC that can be connected to a PC screen or TV and uses a standard console and mouse. The Computer the size of a credit card is Raspberry Pi. You need to plug in a keyboard, mouse, display, power supply, SD card, and a system software that is already installed. The Raspberry Pi is an affordable 8-bit microcontroller that can complete a variety of crucial functions. It can run as a small PC, a portable computer which is capable of coding, a hub for homebrew hardware, and much more. Contains GPOI (general purpose input/output) pins for controlling electronics. It's also a great machine to help kids learn how the computer works, improve their programming skills, and nurture the next generation of developers. This project uses the latest version of the '*Raspberry Pi 3 Model B+*' which has a more powerful quad-core processor. The Raspberry Pi 3 officially uses a 5.1 V micro USB power supply. It comes with 4 G.B. R.A.M. It offers performance that is 3x better than previous versions. In addition it provides 2 micro HDMI ports, Micro SD card slot, Display Serial Interface (DSI), Camera Serial Interface (CSI), and convenient storage.



Fig. 3: Raspberry Pi 3 Model B+

B. *Raspberry Pi Camera*

The Raspberry Pi Foundation created the Raspberry Pi Camera Module. The camera on a Raspberry Pi board can connect directly to the Raspberry Pi's CSI (Camera Serial Interface) connector. The 3.3V power up the ribbon cable is permanently applied for Camera. It features 5 MP resolution with an Omni vision sensor in a fixed focus module. Connect them to the The Raspberry Pi Foundation created the Camera Module for the Raspberry Pi. Camera made using a Raspberry Pi. It is small and weighs about 3g. The purpose of the camera is to capture images and save them to the Raspberry Pi.



Fig. 4: Raspberry Pi Camera

C. *Ultrasonic Sensor*

An A transducer known as an ultrasonic sensor makes use of the physical features and several additional impacts of ultrasonic waves that can send or receive

ultrasonic signals of a certain strength at a certain frequency. They come in piezo or electromagnetic versions. Piezo type is typically selected since it is less expensive and easier to use than other varieties.[7]. The model is primarily based on the ultrasonic distance sensor principle, or simply the ultrasonic sensor. With a range precision that can reach up to 3mm, this affordable sensor offers non-contact measurement capability from 2 cm to 400 cm (about 13 feet). It uses 5 volts to work. It works with 40 kHz ultrasound and when triggered by the transmitting module, the receiving module receives an echoing of the activated signal with a detection tilt of 30°.



Fig. 5: Ultrasonic Sensor

D. *Headphones*

The headphones are used to help the blind person to hear what is the detected object. The headphones are connected to the Raspberry Pi circuit and help the blind person to easily detect the object by hearing the sound.

Sr No.	Component	Approx. Cost
1.	Raspberry Pi 3	Rs. 5000/-
0.	Ultrasonic Sensor	Rs. 80/-
0.	SD Card	Rs. 250/-

0.	Raspberry Pi Cam	Rs 300/-
0.	Headphones	Rs. 200/-
	Glasses	Rs. 300/-
	Total	Rs. 6130/-

Table 2: Product Cost Estimation

2. Software Requirement

a. *Raspberry Pi Imager*

Raspberry Pi Imager is a tool developed by Raspberry Pi Foundation that permits you to easily install an operating system (OS) on your Raspberry Pi step-by-step. A one-stop shop for downloading, configuring, SD card formatting, and operating system flashing for the operating system of your choice. All you need is an SD card reader to write your operating system to an SD card. Most modern computers have a built-in SD card reader.

b. *MS Visual Studio Code*

Visual Studio Code, also commonly known as Microsoft, has developed the source programming environment VS Code for Windows, Linux, and macOS leveraging the Electron Framework. Debugging, syntax highlighting, intelligent code finishing, excerpts, script refactoring, and built-in Git compatibility are among the features.

c. *Python 3.10.6*

Python 3.10.6 is the latest major release of the Python programming language and includes many new features and optimizations.

d. *Raspbian Operating System*

For such a Raspberry Pi line of affordable single-board computers, the Raspberry Pi OS is a Linux operating system that relies on the Debian Version of linux.

e. ***Git Version Control***

Git is a version control system, it is widely used by the developers all over the world. It helps multiple developers to collaborate and work together remotely.

6. Results and Discussion

```
Object Detected at 22 centimeters
INITIATING CAMERA DETECTION...
Object Detected at 215 centimeters
Object Detected at 214 centimeters
Object Detected at 281 centimeters
Object Detected at 140 centimeters
INITIATING CAMERA DETECTION...
Object Detected at 211 centimeters
Object Detected at 217 centimeters
Object Detected at 134 centimeters
INITIATING CAMERA DETECTION...
Object Detected at 1212 centimeters
Object Detected at 290 centimeters
Object Detected at 289 centimeters
Object Detected at 209 centimeters
Object Detected at 74 centimeters
INITIATING CAMERA DETECTION...
Object Detected at 26 centimeters
INITIATING CAMERA DETECTION...
```

Fig 6: UltraSonic Sensor Distance Approximation



Fig 7: Prototype

As seen in Fig 6, The ultrasonic sensor is able to detect objects rapidly. For objects that are not in the declared threshold range (150 cm), the system is not initiating the Camera module thus saving computational power and allowing the system to function for a long time. As for objects falling within a distance of 150 cm, the system would initiate a camera detection module in order to detect and recognize objects in real time.

Fig 7. is the prototype that we have designed, where the Raspberry pi, ultrasonic sensor, pi cam are mounted on the glasses.

7. Conclusion and Future Scope

In conclusion, these smart glasses are especially designed for visually impaired people. So that they can have a more confident and independent life. This glass can help them move around safely as it can detect nearby objects. In our upcoming work, we'll add more capabilities to the text proposed technique and handle the ui problems that come with text comprehension for blind users. Also converting the text to local language which would help people who know only a specific language. This portable device does not require internet connection and can be used independently by people.

8. References

1. https://nei.nih.gov/news/scienceadvances/discovery/project_prakash.
<http://www.who.int/mediacentre/factsheets/fs282/en/>
2. . Rohit Sheth, Surabhi Rajandekar, Shalaka Laddha and Rahul Chaudhari (2014). American Journal of Engineering. [Online]. 03(10), pp-84-89. Available: [http://www.ajer.org/papers/v3\(10\)/L031084089.pdf](http://www.ajer.org/papers/v3(10)/L031084089.pdf)
3. G.Gayathri,M.Vishnupriya,R.Nandhini,Ms.M.Banupriya(2014, March). International Journal of Engineering And Computer Science. [Online]. 03(03), pp-4057-4061. Available: <http://ijecs.in/issue/v3-i3/8%20ijecs.Pdf>
4. Sarah Griffiths and Fiona Macrae. (2014, June). Smart glasses for the BLIND: Display turns the world into outlines to help people with poor vision 'see' obstacles and faces.[Online].Available:<http://www.dailymail.co.uk/sciencetech/article-2659993/Smart-glasses-BLIND-Device-transforms-world-outlines-shapes-help-partially-sighted-navigate.html>
5. Steve (2010, Dec). HALO-Haptic Feedback System for Blind/Visually-Impaired. [Online]. Available: <http://www.polymythic.com/2010/12/teaser-haptic-feedback-for-visually-impaired>
6. K. Shrivastava, A. Verma, and S. P. Singh. Distance Measurement of an Object or Obstacle by Ultrasound Sensors using P89C51RD2.(2010,February).International Journal of Computer Theory and Engineering.[Online].02(01), pp-1793 - 8201. Available: <http://ijcte.org/papers/118-G227.pdf>
7. I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
8. https://ijirt.org/master/publishedpaper/IJIRT151189_PAPER.pdf
9. <https://turcomat.org/index.php/turkbilmec/article/view/1979>
10. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
11. <https://www.sciencedirect.com/science/article/pii/S2666990021000148>
12. R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), 2015.

13. <https://blog.tensorflow.org/2021/01/custom-object-detection-in-browser.html>
14. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
15. <https://www.irjet.net/archives/V7/i6/IRJET-V7I6577.pdf>.
16. S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, “Learning Rich Features from RGB-D Images for Object Detection and Segmentation,” Computer Vision – ECCV 2014 Lecture Notes in Computer Science, pp. 345–360, 2014.
17. <https://www.irjet.net/archives/V7/i5/IRJET-V7I5854.pdf>
18. <https://circuitdigest.com/microcontroller-projects/arduino-ultrasonic-sensor-based-distance-measurement>
19. <https://core.ac.uk/download/pdf/83592918.pdf>
20. <https://www.ijert.org/implementation-of-text-to-speech-conversion>
21. “An introduction to Text-To-Speech in Android,” Android Developers Blog, 23-Sep-2009. [Online]. Available: <https://androiddevelopers.googleblog.com/2009/09/introduction-totexttospeech-in.html>. [Accessed: 24-Mar-2019]
22. Bhardwaj, R. J., & Rao, D. (2022). Deep Learning-Based Traffic Behavior Analysis under Multiple Camera Environment. International Journal of Next-Generation Computing, 13(3).
23. Bhardwaj, R. J., & Rao, D. (2022). Modified Neural Network-based Object Classification in Video Surveillance System. International Journal of Next-Generation Computing, 13(3).

Automated Histogram Binning Based Fuzzy K-Means Clustering for COVID-19 Chest CT Image Segmentation

S. Nivetha¹ and H. Hannah Inbarani^{2*}

¹Department of Computer Science, Periyar University, Salem, TamilNadu, India.
nivethas@periyaruniversity.ac.in

²Department of Computer Science, Periyar University, Salem, TamilNadu,
India.hhinba@periyaruniversity.ac.in

Abstract. The greatest threat to humanity is COVID-19, which has a global impact on billions of people. For important judgment and disease control, therapeutic imaging, such as Computed Tomography (CT), has a lot of potential as an alternative to the Real Time Reverse Transcription–Polymerase Chain Reaction (RT-PCR) test. Automatic image segmentation is therefore highly sought after as a decision support system. Image dissection is dividing a figure into sections entrenched on a set of criteria. In this study, a dataset on COVID CT scan is analyzed using the proposed method of Histogram Binning Based on Fuzzy_K Means_Clustering (HBFKM) with the existing two main cluster methods namely Fuzzy_K-Means (FKM) and Possibilistic_Fuzzy_C-Means (PFCM) and are utilized throughout the segmentation step to segment the images. The findings specify that related to the other approaches under study, the suggested method Histogram Binning Based on Fuzzy K Means Clustering offers the highest accuracy and reliability with 85.08 percent and 85.28 percent precision. Additionally, the outcomes demonstrate that the proposed approach has the maximum accuracy with a specificity rate of 85.18 %. ratio. And lastly, the proposed technique outperforms the others with an F1-score rate of 85.47 percent.

Keywords: COVID-19, Histogram Binning Based Fuzzy K Means Clustering, Fuzzy_K-Means (FKM), Possibilistic_Fuzzy_C-Means(PFCM), Image Segmentation.

1. Introduction

Segmentation is the division of an image's pixels into nonoverlapping, regular areas that seem to be identical regarding certain factors relating to grey-level luminance or appearance is known as segmentation. There are several key problems in the field of image processing that are considered heavily researched areas. With consideration of machine learning methodologies, each level of image segmentation is required and plays a role in some of the many applications in the study of computer sight and image processing. Throughout the present and near future,

image data will continue to grow in both volume and impact. Image processing techniques are commonly used when working with image data and are considered necessary tools. Medical imaging is one of the world's most ground-breaking medical breakthroughs. Medical diagnosis and intervention create a visual representation of the interior of the body. Medical imaging techniques, however, do not create flawless images, and many aberrations, such as noise, and the complex architecture of particular biological tissues. The Fuzzy_K-Means (FKM) algorithm is incredibly effectively utilized data similarity techniques.

Image segmentation is based on two fundamental image properties. One is discontinuity, which denotes quick or sudden vicissitudes in luminance as edges, and the other is similarity, which states grouping digitized imagery into areas based on certain preconceived resemblance conditions. Image analysis takes recently sparked a lot of attention, and it's becoming increasingly important. In this aspect, Histogram_Thresholding [1], Edge_Detection [2], Region_Growing [3], Watershed_Transformation [4], Clustering, and Soft_Computing approaches have all been investigated in the literature. Image segmentation utilizing Genetic algorithms [5], Fuzzy_Logic_techniques [6], and Neural_Network_based approaches [7], are all instances of Soft Computing techniques.

2. Related Work

Dhruv, B. et al. [8] proposed Hybrid_Particle_Swarm_Optimised_Fuzzy-C_Means Clustering which was used to develop a hybrid technique for effective COVID-19 screening utilizing chest CT images. The suggested method was evaluated on 15 individuals with torso CT images of COVID-19 infection and the findings were quantitatively confirmed using metrics including entropy, contrast, and standard deviation, revealing that it exceeds traditional Fuzzy C Means Clustering. In [9], SUFMACS (Superpixel based_Fuzzy_Memetic_Advanced_Cuckoo_Search) is a unique unsupervised machine learning-based method for quickly analyzing and sectioning COVID-19 medical images. This method adapts the superpixel method to compress vast spatial data. The Luus-Jaakola heuristic method is merged with McCulloch's methodology, and the original cuckoo search method is modified. The fuzzy modified objective function is optimized using cuckoo search method. This objective function takes advantage of the superpixel advantages. The Fast Fuzzy C Mean Clustering approach is suggested in [10] for the purpose of COVID-19 disease prediction in CT scan images. The ROI extractor result generated by the Fast FCM technique, which is focused on histogram evaluation, are efficient and have low computational complexity. Prior to RT-PCR lab testing, CT is regarded as a significant COVID-19 diagnostic approach, and ROI extraction is becoming more crucial. Compared to the results of the traditional

segmentation algorithm, the Fast Fuzzy C Means Clustering methodology is an automated ROI extraction method. In [11], study is an image-processing-based technique for diagnosing COVID-19. The proposed Fuzzy C-Means (FCM) is then employed as a novel approach for segmenting COVID-19 chest X-rays. The system classification, using the Enhanced Capsule Network (ECN), was accomplished. The method relies on deep learning and has different levels, with the first stage being the processing of original information and the last being the presentation of the model categorization. To enhance the ECN, the MayFly Optimization (MFO) method was applied. To make the COVID-19 radiological images clearer, Shouvik Chakraborty et al. [12], presented a special approach for segmenting radiological images. The suggested strategy combines a type 2 fuzzy segmentation method with SuFMoFPA (Superpixel-based Fuzzy Modified Flower Pollination Algorithm) to provide a more segmented outcome. The idea of pixels in the image makes the process simpler, eliminating the need for processing a significant quantity of positional information. This proposed method is used with a type 2 fuzzy system to produce a more differentiated outcome. The suggested strategy might involve using a computer-aided device to stop the coronavirus growth. Table 1 depicts the summary of related works.

Table 1. Summary of related work

S.N O	MO DAL ITY	AUT HOR S	DATASET	METHOD	OUTCOMES AND METRICS
[8]	CT	Dhruv , B. et al.	15 Chest CT Imageries	Hybrid_ Particle_Swarm _Optimized and Fuzzy-C Means_Clustering_ based segmentation.	COVID 19 Lesion Image. Metrics are entropy, contrast and standard deviation
[9]	X-Ray and CT	Chakraborty , S., et al.	Among COVID-19-positive individuals in Italy, China, Iran, Taiwan, Korea, and other	SUFMACS (SUperpixel-based Fuzzy Memetic Advanced Cuckoo Search)	COVID 19 Segmented Image. Based on the number of cluster calculated metrics are Davies–Bouldin index , Xie-Beni index ,

			nations, 250 CT and 250 X-ray images were collected.		Dunn index , β index
[10]	CT	Kumar, S. N. et al.	http://corona cases.org ,	Fast_Fuzzy_C means clustering algorithm.	Automatic ROI extraction. Measures are Partition Coefficient (PC) and Partition Entropy (PE) measures.
[11]	X-Ray	Farkhi, A., Salek shahrezaee et al.	Wuhan University's Renmin Hospital and two important hospitals in Guangzhou, Sun Yat-sen Memorial Hospital and Third Affiliated Hospital of Sun Yat-sen University with 12 and 76 patients, respectively.	Enhanced variation of the Enhanced_Capsule_Network(ECN) and an improved Fuzzy _C-Ordered_Means (FCOM).	Segmented Image. Metrics are 97.08% Accuracy 97.29% Precision, 97.18% ,Sensitivity,97.47 % F1-Score
[12]	CT and X-Ray	Shouvik Chakr	115 chest-related CT scanning results.	Superpixel-based_Fuzzy _Modified_Flower	Segmented COVID-19 Image. Metrics are

		aborty et al.		Pollination_Algorithm.	Davies–Bouldin index , Xie-Beni index ,Dunn index β index. Based on the number of cluster
--	--	---------------	--	------------------------	--

3. Fuzzy K Means Clustering

Dunn invented Fuzzy K-means clustering in 1973 as a member of binary or more sets of data elements [13]. This benchmark is commonly used in image analysis and pattern recognition images from medical, geological, and satellite sources, etc., The FKM algorithm is primarily concerned with reducing the estimates of the objective function. The objective function evaluates the cluster's quality divides the data and splits it into c clusters.

Concerning some given criterion, the FKM algorithm tries to divider a fixed group of n component $X = \{x_1, \dots, x_n\}$ into a gathering of c fuzzy clusters. The algorithm returns a partition matrix and a listing of c group midpoints $U = u_{i,j} \in [0,1], i = 1,2, \dots, n; j = 1,2, \dots, c$ then $C = \{c_1, \dots, c_c\}$, where each element u_{ij} indicates the degree which element x_i is a member of cluster c_j . In hard clusters, an object's membership in a cluster is Boolean. In other words, the object both fits to the cluster or it does not. Rather, fuzzy clustering enables every item to belong to a number of clusters, with degree of membership ranging from 0 to 1, depending on the distance between the object and the cluster centres. The objective function is used continually in an attempt to minimize the distance. The following objective function,

$$J = \sum_{i=1}^n \sum_{j=1}^K U_{ij}^m \|x_i - c_j\|^2 \quad 1 \leq m < \infty \quad (1)$$

where K is the number of clusters, U_{ij} is the degree of participation of pixel x_i in the i^{th} group, c_i is the cluster j's centroid, and $\|\cdot\|$ is the Euclidean distance between each pixel and the centroid, m is a constant value termed fuzzifier that exists and regulates the solution set's fuzziness. Fuzzy K-Means Clustering is explained in [14]. Fuzzy partitioning is accomplished by iteratively optimizing the objective function described above, with membership u_{ij} and cluster centers v_i updated as follow,

$$u_{ij} = \frac{1}{\sum_{j=1}^K \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad i = 1, 2, \dots, n \quad (2)$$

The membership functions and cluster centroids are updated using,

$$c_i = \frac{\sum_{j=1}^M u_{ij}^m x_j}{\sum_{j=1}^M u_{ij}^m} \quad (3)$$

4. Possibilistic Fuzzy C-Means Clustering

A work by Pal [15] suggested novel technique called Possibilistic Fuzzy C-Means (PFCM), which combined PCM and FKM and solved numerous issues with PCM, FKM, and PFCM. When executing classification tests, J. C. and N. R. Bezdek Pal proposed PFCM as a good clustering technique since it can influence the membership value or canonical value significantly. The noise sensitivity is determined by PFCM. Possibilistic Fuzzy C-Means Clustering is explained in [14].

The usage of PFCM has been widespread in many different areas [16],[17] such as “shell clustering”, “boundary detection”, “surface and function approximations” and has successfully solved a number of problems. Because a and b had been actually fixed to 1, it may be concluded that while computing centroids, both belonging and probability were equally important. Certain clustering conclusions lost some of their clarity because PFCM incorporated variables a and b for belonging and probability, that showed the key importance in the formation of cluster centers.

Assume the following unlabeled data sets: $X = \{x_1, \dots, x_n\} \subset \mathbb{R}^p$ ($p = n \times s$) will be grouped into c ($1 < c < n$) clusters, a fuzzy subset. Here, n denotes the total number of data points, and s , their individual dimensions. The following objective functions are minimised to meet the goal of clustering X into c clusters.

$$J = \sum_{k=1}^n \sum_{i=1}^c (\alpha u_{ik}^m + \beta t_{ik}^\tau) d_{ik}^2 + \sum_{i=1}^c \delta_i \sum_{k=1}^n (1 - t_{ik})^{-\tau} \quad (4)$$

Parameters α and β determine the relative importance of membership values and typicality values. t_{ik} determines possibilistic membership degree. The Euclidean distance between the j -th data and the i -th cluster centre vector is given by the formula $d_{ik}^2 = \|x_k - v_i\|^2$. Cluster centre vector is stated by $V = \{v_1, v_2, \dots, v_c\}$, $v_i \in \mathbb{R}^s$, where $\delta_i > 0$ is the typical possibilistic value. Here, the weighting exponents $m > 1$ and $\tau > 1$ are used. Fix $m > 1$, $\tau > 1$, $\epsilon > 0$, and $1 < c < n$. Pick $v^{(0)} \in \mathbb{R}^s$, $v^{(0)}$ can be preferred arbitrarily since $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^p$. Using the

following, determine the fuzzy membership degree (u_{ik}), which decreases the objective function $J_{m,\tau}$.

$$u_{ik}^{(l)} = \left(\sum_{j=1}^c \left(\frac{d_{jk}^2}{d_{ik}^2} \right)^{\frac{1}{m-1}} \right)^{-1} \quad (5)$$

With the following formula, determine the probabilistic typical δ_i that minimizes the objective function $J_{m,\tau}$.

$$\delta_i^{(l)} = \frac{\sum_{k=1}^n (u_{ik}^{(l)})^m d_{ik}^2}{\sum_{k=1}^n (u_{ik}^{(l)})^m} \quad 1 \leq k \leq n \quad (6)$$

With the following formula, determine the probabilistic typical t_{ik} that minimizes the objective function $J_{m,\tau}$.

$$t_{ik}^{(l)} = \left(1 + \left(\frac{\beta}{\delta_i} d_{ik}^2 \right)^{\frac{1}{\tau-1}} \right)^{-1} \quad 1 \leq i \leq c; 1 \leq k \leq n \quad (7)$$

Use the following to update cluster center v_i , which minimizes the objective function $J_{m,\tau}$. Apply $v_i^{(l)}$ to $v_i^{(l-1)}$ using $\|v_i^{(l)} - v_i^{(l-1)}\| < \epsilon$. If it's correct, stop. Otherwise, set $l = l + 1$

$$v_i^{(l)} = \frac{\sum_{k=1}^n ((\alpha u_{ik}^{(l)})^m + (\beta t_{ik}^{(l)})^\tau) x_k}{\sum_{k=1}^n ((\alpha u_{ik}^{(l)})^m + (\beta t_{ik}^{(l)})^\tau)} \quad 1 \leq k \leq n \quad (8)$$

5. Proposed Histogram Binning Based Fuzzy K Means Clustering (HBFKMC)

The separation of objects within an image might be mentioned as segmentation. Clustering is generally employed to identify boundaries or objects within an image. Defining and examining the image at the end of the process would be simpler. Nonparametric density estimators, such as histograms, are widely used for data visualization and obtaining summary numbers. An image's histogram, however, displays the frequency of pixel intensity values. While segmenting images using the fuzzy K-means clustering approach, the quantity of groups into which the image is to be segmented must be specified to make the segmentation process automatic. The quantity of groups in an image can be determined by looking at the histogram of the image. Fig 1 depicts the proposed methodology for Histogram Binning Based on Fuzzy K Means Clustering. Algorithm 1 depicts the pseudocode

for Histogram Binning Based on Fuzzy K Means Clustering. In this work, number of bins is computed automatically based on the structure of the image using Struge's rule.

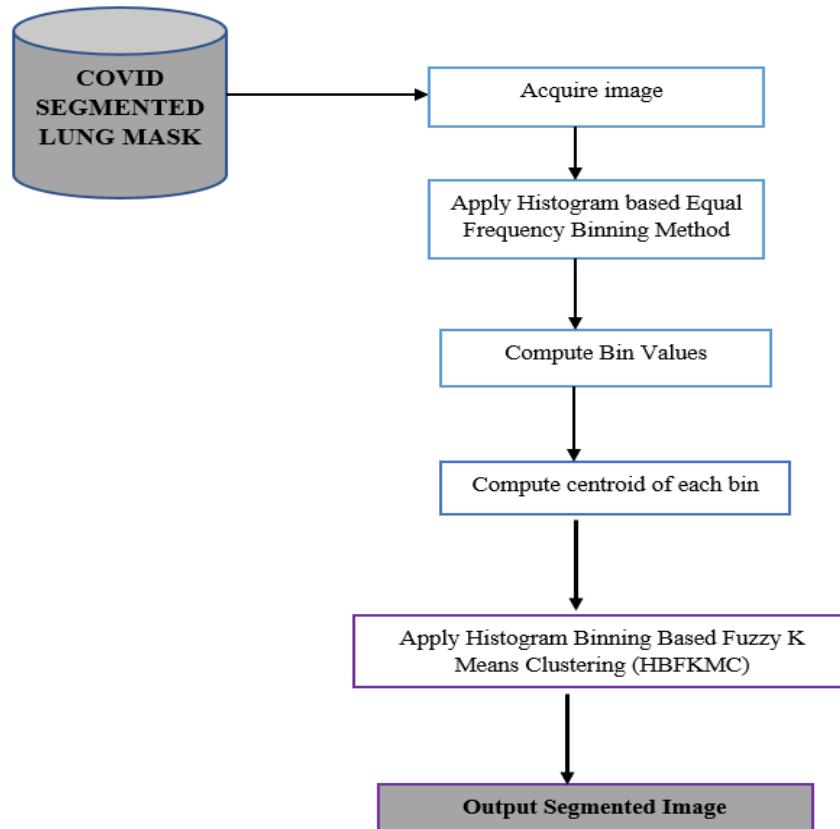


Fig. 1. Proposed methodology for HBFKMC

6. Performance Measure of Segmentation

To estimate the effectiveness of performance of each algorithm, the ratios of successfully recognised objects and regions are used. Several metrics are used to determine the efficiency across an entire set of images and then to measure how efficient the segmentation process is. These metrics include Accuracy, Precision, Recall, Specificity, and F1-Score[18],[19],[20].

INPUT: Segmented Mask Image, $D(x,y)$ = Input_Image, K=Bin_Value, m =Fuzzification Value, ϵ = Threshold
OUTPUT: Clustered Image

STEP 1 : Input Segmented Mask Image as source D(x,y) Image.

STEP 2 : Apply Histogram Based Equal Frequency Binning Method.

STEP 3: Compute Number of Bins using Struge's Rule , $K = 1 + 3.322 \log_N$

K = Number of Bins , N =Number of Observations , Log = Logarithm of N

STEP 4: Create Histogram Width Frequency Binning (K=BINS=Number of Clusters)

STEP 5 : Obtain number of bins K from the histogram.

STEP 6: Compute each bin's centroid value and set them as cluster centroids.

$$C = \{c_1, \dots, c_k\}$$

STEP 7: Initialize Membership u_{ij}

$$u_{ij} = \frac{1}{\sum_{j=1}^K \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{m-1}}$$

STEP 8 : Compute Distance $\|x_i - c_j\|$ between each pixel and the centroid.

$$J = \sum_{i=1}^n \sum_{j=1}^K U_{ij}^m \|x_i - c_j\|^2$$

STEP 9: Update Membership

STEP 10 : Update Cluster Center

$$v_i = \frac{\sum_{j=1}^M U_{ij}^m x_i}{\sum_{j=1}^M U_{ij}^m}$$

STEP 11 : Update Distance between clusters centers and pixels

STEP 13 : Compute Objective Function as J'

STEP 14 : If $\|J - J'\| < \epsilon$ Until the condition is satisfied.

STEP 15 : Reshape cluster pixels into an image.

Factors are used to evaluate the algorithm and the equations are given beneath,

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (9)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (10)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (11)$$

$$\text{Specificity} = \frac{TN}{FP+TN} \quad (12)$$

$$\text{F1 - Score} = \frac{2 * \text{PRECISION} * \text{RECALL}}{\text{PRECISION} + \text{RECALL}} \quad (13)$$

Where TP refers to True Positive; FP refers to False Positive; FN refers to False Negative; TN refers to True Negative.

Structural Similarity Index Method

The structural similarity index is a method for determining exactly similar two images seem [21].

$$\text{SSIM}(A, B) = \frac{(2\mu_A\mu_B + C_1)(2\sigma + C_2)}{(\mu_A^2 + \mu_B^2 + C_1)(\sigma_A^2 + \sigma_B^2 + C_2)} \quad (14)$$

where μ_A and μ_B denotes the mean values of original and distorted images. And σ_A and σ_B denotes the standard deviation of original and distorted images, and σ_{AB} is the covariance of both images and C_1 and C_2 are constants. For the proposed methodology the loss of visual structure, brightness, contrast, and structure similarity are measured. SSIM can serve as a second opinion to boost radiologists self-assurance.

7. Results and Discussion

Anaconda3 is used to implement the proposed Histogram Binning Based on Fuzzy K Means Clustering (HBFKMC) approach. For algorithm study and evaluation, medical images of COVID have been considered source images. The dataset is available at the GitHub repository [https://github.com/UCSD-AI4H/COVID-CT\[22\]](https://github.com/UCSD-AI4H/COVID-CT[22]). The performance of the three algorithms Proposed HBFKM, FKM, and PFCM is shown and compared in this section in terms of the findings. The values have been set to modest values in several sets of original images: $\alpha = 1$, $\beta = 4$, $nc = 2$ and $\text{max_iter} = 1000$. In comparison to the traditional algorithms Fuzzy K-Means Clustering (FKM), and Possibilistic Fuzzy C-Means (PFCM) that are frequently employed for processing medical images and clustering fields, the results of the proposed approach are shown in this part. The comparison compares the

methods using COVID images and five metrics: Accuracy, Precision, Recall, Specificity, and F1-Score[23],[24]. As shown in Table 2, the FKM and PFCM only reached 76.38 and 79.18 percent of the Precision respectively, and recall 76.03 and 79.02 for FKM and PFCM. It is evident from Fig. 2 that the Proposed HBFCMC performs better at classifying attacks than the current segmentation algorithm. The average of precision, recall, and F-measure for all classes is used to specify the quantitative outcomes of precision, recall, and F-measure.

Fig 3 shows the original image in the first column, the proposed clustered image in the second column to help determine how many clusters there are as (K), the clustered image in the third column, and the extracted lungs in the fourth column and infection of COVID-19 in the last column. It has been noted that better segmentation results are obtained in the proposed segmentation method..The original images and the segmented outcomes of the FKM and PFCM are shown in Fig 4 and 5. To present some of the images, the images are separated into figures as Clustered images, extracted lungs, and infection of the lungs as separate figures. These outcomes demonstrate that the proposed model may effectively slice the chest CT imageries using COVID-19.

Table 2.Comparison of the proposed approach with standard Fuzzy C-Means algorithm and PFCM on segmented images.

	Proposed HBFKMC	FKM	PFCM
Accuracy	85.08	76.05	79.00
Precision	85.28	76.38	79.18
Recall	85.00	76.03	79.02
Specificity	85.18	76.25	79.16
F1-Score	85.47	76.57	79.67

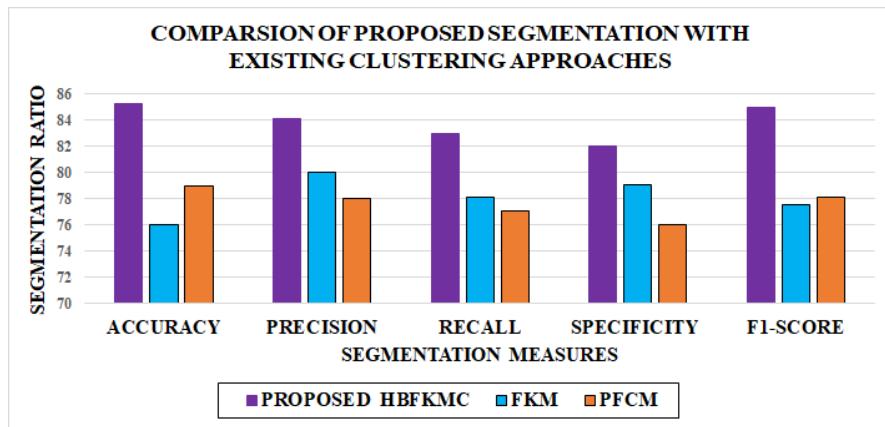


Fig. 2. Relative quantitative measures for COVID-19 Segmentation

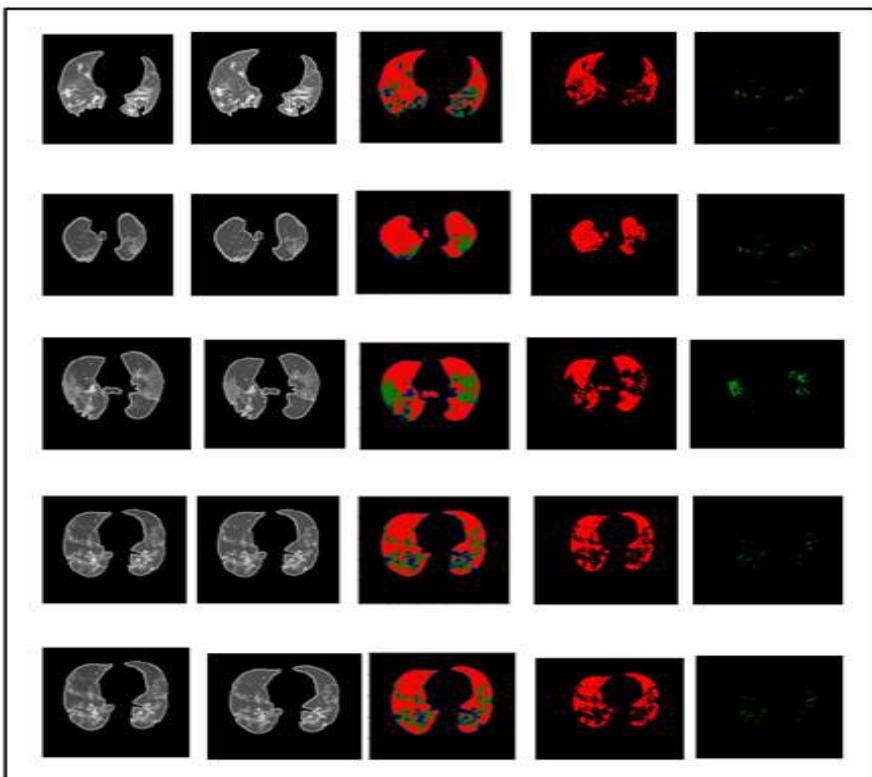


Fig. 3. Segmentation of the proposed HBFKMC for COVID images.a) Actual imagery
b)Clustered picture c) Segmented picture d)Segmented lung picture e)Infection picture

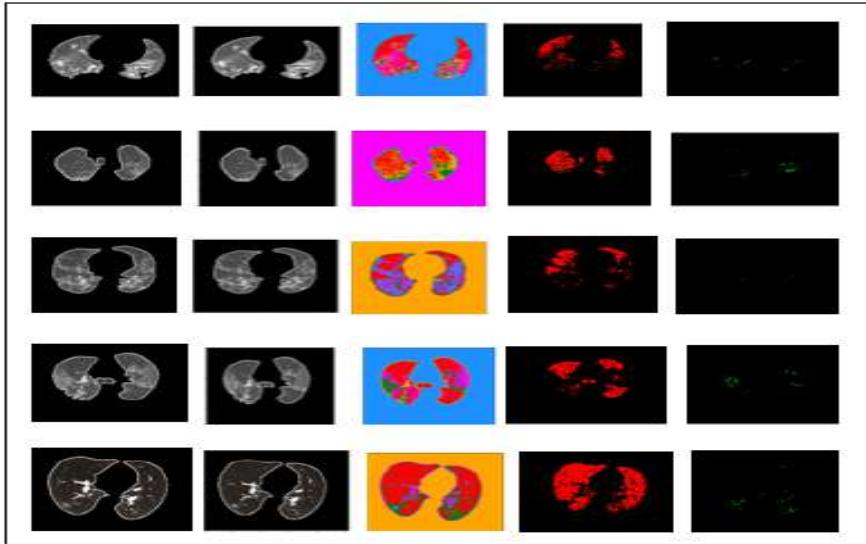


Fig. 4. Segmentation of the Fuzzy K-Means Clustering for COVID picture. a) Actual image b) Clustered picture c) Segmented picture d) Segmented lung picture e) Infection picture

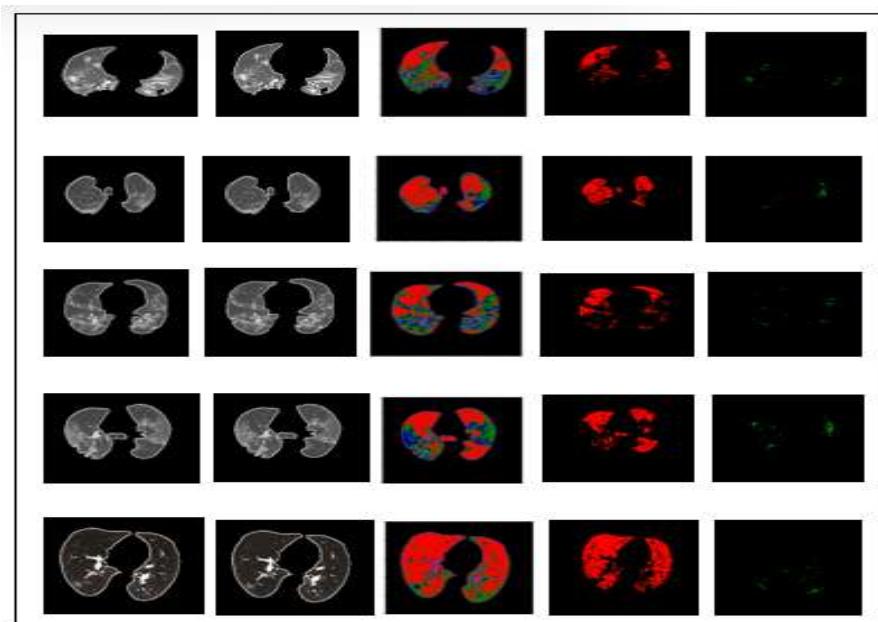


Fig. 5. Segmentation of the Possibilistic Fuzzy C-Means Clustering for COVID images.a) Actual Picture b) Clustered picture c) Segmented picture d) Segmented lung picture e) Infection picture

6. Conclusion

This study suggested a technique for segmenting lung infection regions from a CT volume of a COVID-19 patient. In this study, the effectiveness of three machine learning approaches Histogram Binning Based on Fuzzy K Means Clustering (HBFKM), the Fuzzy K-Means (FKM), and Possibilistic Fuzzy C-Means (PFCM) was evaluated for the detection of diseased regions in images of the lungs taken from COVID-19 patients. The outcomes Histogram Binning Based on Fuzzy K Means Clustering (HBFKM) showed that the was able to distinguish between unhealthy and infected tissues in these images. The supplied algorithm's performance was assessed using metrics including accuracy, sensitivity, F-measure, precision, specificity, metrics including the supplied algorithm's performance was assessed. The proposed method, followed by FKM and PFCM respectively, had the greatest accuracy 85.08% in COVID images. This indicates that, compared to other algorithms, the proposed method may achieve the highest accuracy rate with the original images. The proposed work can measure the COVID-19 lesion, depict the infected region, and promptly help assess changes in the images. Additionally, the suggested method can identify abnormal areas with minimal contrast between lesions and healthy tissues.

Acknowledgement

The authors would like to thank UGC, New Delhi, for the financial support received under UGC-SAP No. F.5-6/2018/DRS-II (SAP-II).

References

1. S. K. Pal, R. A. King, and A. A. Hashim, "Automatic grey level thresholding through index of fuzziness and entropy," *Pattern Recognit.Lett.*, vol. 1, pp. 141–146, Mar. 1983.
2. Batista, J., Freitas, R. An adaptive gradient-based boundary detector for MRI images of the brain, *Image Processing And Its Applications*, Seventh International Conference, vol.1, pp.440 – 444, Jul 1999.
3. J.P. Fan, D.K.Y Yau, and A.K. Elmagarmid. Automatic image segmentation by integrating color edge extraction and seeded region growing. *IEEE Transactions on Image Processing*, 10(10):1454–1466, 2001
4. A. Bleau and L. J. Leon, "Watershed-based segmentation and region merging," *CVIU*, vol. 77, no. 3, pp. 317–370, Mar. 2000
5. M. Abdulghafour, "Image segmentation using Fuzzy logic and genetic algorithms", *Journal of WSCG*, vol.11, no. 1, 2003
6. Kanchan Deshmukh and G. N. Shinde, "An adaptive neuro-fuzzy system for color image segmentation", *J. Indian Inst. Sci.*, vol. 86, Sept.-Oct. 2006, pp.493-506.
7. Jander Moreira and Luciano Da Fontoura Costa, "Neural-based color image segmentation and classification using self-organizing maps", *Anais do IX SIBGRAPI*, 1996, pp.47-54.

8. Dhruv, B., Mittal, N., & Modi, M. (2022). Hybrid Particle Swarm Optimized and Fuzzy C Mean Clustering based segmentation technique for investigation of COVID-19 infected chest CT. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 1-8.
9. Chakraborty, S., & Mali, K. (2021). SUFMACS: A machine learning-based robust image segmentation framework for COVID-19 radiological image interpretation. *Expert Systems with Applications*, 178, 115069.
10. Kumar, S. N., Ahilan, A., Fred, A. L., & Kumar, H. A. (2021). ROI extraction in CT lung images of COVID-19 using Fast Fuzzy C means clustering. In *Biomedical engineering tools for management for patients with COVID-19* (pp. 103-119). Academic Press.
11. Farki, A., Salekshahrezaee, Z., Tofigh, A. M., Ghanavati, R., Arandian, B., & Chapnevis, A. (2021). COVID-19 Diagnosis Using Capsule Network and Fuzzy-Means and Mayfly Optimization Algorithm. *BioMed Research International*, 2021.
12. Chakraborty, S., & Mali, K. (2021). SuFMoFPA: A superpixel and meta-heuristic based fuzzy image segmentation approach to explicate COVID-19 radiological images. *Expert Systems with Applications*, 167, 114142.
13. Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & geosciences*, 10(2-3), 191-203.
14. Yang, Y., Zheng, C., & Lin, P. (2005). Fuzzy C-means clustering algorithm with a novel penalty term for image segmentation. *Optoelectronics Review*, 13(4), 309.
15. N. R. Pal, K. H. Pal, and M. K. James, "A possibilistic fuzzy c-means clustering algorithm," IEEE TFS, vol. 13, no. 1, pp. 517–530, 2005
16. C. L. Chowdhary and D. P. Acharya, "Segmentation of mammograms using a novel intuitionistic possibilistic fuzzy c-mean clustering algorithm," Nature Inspired Computing, vol. 652, pp. 75–82, 2018
17. S. Askari, N. Montazerin, M. H. F. Zarandi, and E. Hakimi, "Generalized entropy-based possibilistic fuzzy C-Means for clustering noisy data and its convergence proof," Neurocomputing, vol. 219, pp. 186–202, 2017
18. Gite, S., Mishra, A., & Kotecha, K. (2022). Enhanced lung image segmentation using deep learning. *Neural Computing and Applications*, 1-15.
19. Monteiro, F. C., & Campilho, A. C. (2006, September). Performance evaluation of image segmentation. In *International Conference Image Analysis and Recognition* (pp. 248-259). Springer, Berlin, Heidelberg.
20. Nivetha, S., and H. Hannah Inbarani. Neighborhood Rough Neural Network Approach for COVID-19 Image Classification. *Neural processing letters*. Springer. pp. 1-23. (2022).
21. Hore, A., & Ziou, D. (2010, August). Image quality metrics: PSNR vs. SSIM. In 2010 20th international conference on pattern recognition (pp. 2366-2369). IEEE.
22. <https://github.com/UCSD-AI4H/COVID-CT>.
23. Nivetha, S., and H. Hannah Inbarani. Classification of COVID-19 CT Scan Images Using Novel Tolerance Rough Set Approach. In *Machine Learning for Critical Internet of Medical Things*. Springer, Cham. pp. 55-80. (2022).
24. Nivetha, S. and H. Hannah Inbarani (2021). Prediction of COVID-19 fatality cases based on regression techniques. *European Journal of Molecular & Clinical Medicine*, 7(3), 696-719.

An Effective Multi-Exposure Fusion Approach using Exposure Correction and Recursive Filter

*Jishnu C R¹ and Vishnukumar S²

Department of Computer Applications, Cochin University of Science and Technology,

Kochi 682022, Kerala, India

¹jishnudca@cusat.ac.in

²vks@cusat.ac.in

Abstract. The visual quality of a natural scene is constantly degraded when using conventional imaging sensors. The dynamic range is the factor that limits the sensors to capture more information as it is limited to a specific range during capturing. Multi-Exposure Image Fusion (MEF) is a technique for creating a well-exposed, high dynamic range (HDR) like image from a collection of inconsistently exposed, low dynamic range (LDR) images. The fused image may still have image artifacts and lose information, nevertheless. To overcome this, we propose an MEF approach using an effective exposure correction mechanism. The technical strategy for the proposed method is to initially improve the source images using an exposure correction strategy, followed by merging the images using pyramidal decomposition to produce HDR like images. Experimental comparison with existing methods demonstrates that the proposed procedure produces positive statistical and visual outcomes.

Keywords: Multi-Exposure Image Fusion, Exposure Correction, Recursive Filter, Dynamic Range, Exposure Value.

1 Introduction

This work's overarching goal is to create an image that is aesthetically beautiful, rich in details, and has vibrant colors using a few input images. There are a couple of ways to develop a well-exposed highly detailed image starting with illuminating the image. Image illumination, even though the term sounds convincing and easy to perform, the outcome could always deviate from the original scene and could end up being more artificial than natural. Other well-known methods for producing well-exposed images from LDR images include tone mapping [1] and multi-exposure image fusion [2]. Tone mapping operation is often compromised at times due to the fact that the process requires an intermediate HDR image. Due to its ability to develop effective well exposed images, the MEF has claimed wide attention throughout the years. MEF strategies are the ones that provide promising outcomes with more detail and better visual quality, in addition to being hardware independent and not needing the camera response figures. The paper contains a brief study of the existing MEF methods in section 2. A demonstration of the proposed multi-exposure image fusion technique is described on section 3. The comparative study's findings are examined and the results are displayed at section 4. A brief

summary with potential paths for the future work is mentioned in section 5.

2 Literature review

There have been studies on MEF for the past few decades. One of the first researchers in the field of MEF was Burt et al. [3], who presented a pyramid-based approach to carry out the image fusion operation. Then Goshtasby et al. [4] recommended an MEF method in which the input images are separated into uniform blocks, followed by a blending function to blend the ones with more details. However, the object boundaries in the fused images were poor. Mertens et al. [5] proposed an approach in 2007 for fusing a collection of images with variable exposure using straightforward quality indicators like saturation and contrast known as weight terms.

Later on, Liu et al.[6] suggested a method that studies the impact of weight terms on fused images and used a dense scale-invariant transform descriptor for developing the contrast term. An adaptive weights-based MEF technique that represents global gradient and the relative pixel intensity to construct an adaptive rule for the individual images was put forwarded by Lee et al. [7]. Hayat et al. [8] performed a multi-exposure image fusion by pyramidal decomposition of weight maps. In addition to contrast and saturation, a color dissimilarity measure is used to measure the ghost artifacts in the images. A fast-guided filter is used for the weight map refinement.

The approach of Li et al. [9] uses pyramidal decomposition and coefficient attenuation on each level to condense the dynamic range. Fang et al. [10] developed an objective quality model for MEF. Here, similarities between subjects are identified from two alternative forced choices (2AFC), with this an objective quality model is observed for quantifying the ghosting artifacts. Edge-preserving smoothing-based Multi-Scale exposure fusion is a technique Wang et al.[11] designed to preserve the details from the darkest and brightest region from an HDR picture. Karakaya et al.[12], introduced a method that involves the development of a fused image with the combined weight map creation from saliency maps, principal component analysis and adaptive well exposedness, followed by refinement using a guided filter.

The major problem with most of the available MEF approaches is the presence of halo artifacts in the fused image. Other common problems faced by the fused results include color distortions and detail loss. To overcome these problems, we propose

an MEF approach with exposure correction and pyramidal decomposition for more enhanced and prominent fused images.

3 Proposed method

The method begins with evaluating the source images using an Exposure-Value (EV) parameter to identify the underexposed and overexposed images. The exposure correction strategy is implemented on the classified input images to develop the individual exposure-corrected image. Even though the quality of the original image is greatly improved by exposure correction, some areas may still have inconsistencies and color distortions. This is resolved using the recursive filter by refining the original images with their exposure-corrected counterpart, thereby developing an intermediate refined exposure stack. Weight maps are produced by utilizing contrast, saturation, and well-exposedness from the intermediate refined exposure stack. These weights and the refined inputs are then combined to create the fused image using a pyramidal decomposition. The workflow of the proposed MEF approach is shown in Fig. 1.

3.1 Input Refinement using Exposure Correction and Recursive Filter

Here, an effective exposure correction mechanism is used for refining the input images. The objective is to perform respective exposure correction strategies on the underexposed and overexposed images. For this, the input images are initially classified into underexposed and overexposed images based on their intensity exposure. This is accomplished by taking the exposure value (EV) of the input images[13] and designating 0.5 as the threshold, where images with exposure value less than or equal to the threshold are deemed underexposed and images with more than the threshold are considered overexposed.

The image exposure value is determined by

$$\text{exposure value (EV)} = \frac{1}{L} \frac{\sum_{i=1}^L h(i)i}{\sum_{i=1}^L h(i)} \quad (1)$$

L denotes total gray levels whereas $h(i)$ denotes image's histogram.

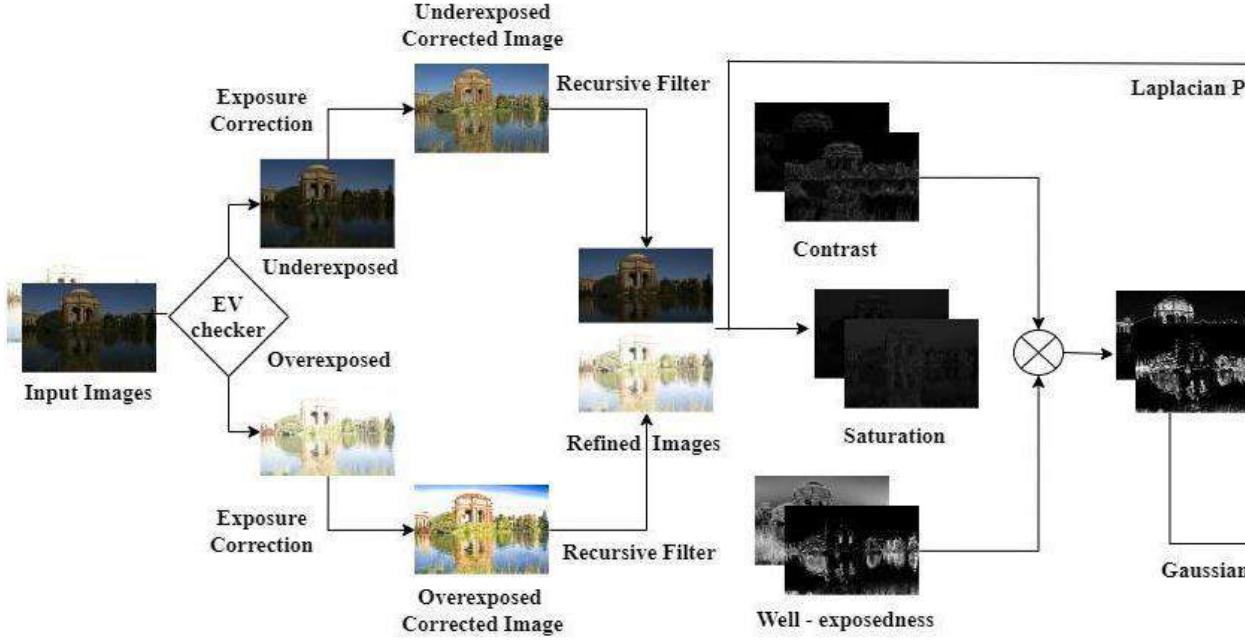


Fig 1. Overview of the proposed MEF approach.

Both underexposure correction and overexposure correction methods adhere to the Retinex-based image enhancement tenet, which states that an image I is determined as the pixel-wise multiplication of its enhanced version I' and the illumination map M [17],

$$I = I' \times M \quad (2)$$

The maximum RGB color channels are used as the illumination values at each pixel of the normalized image to create the initial illumination map,

$$M'_P = \max(I_P^C) \quad (3)$$

where P stands for the pixel and C is the color channel. To prevent the color channel of the recovered image from straying outside of the color gamut, the maximum value of the pixel is chosen. However, because of the additional textures acquired, the initial illumination map can turn out to be inaccurate and unrealistic.

As a result, improving the illumination map is required to preserve the noticeable structure, which is performed using the following objective function:

$$\arg \min \sum_P ((M_P - M'_P)^2 + \lambda(w_{a,p}(\partial_a M)_P^2 + w_{b,p}(\partial_b M)_P^2)) \quad (4)$$

where ∂_a and ∂_b are special derivatives in horizontal and vertical directions respectively, and w is the spatially varying smoothening weights. Together with the spatial derivatives, the redundant details are removed. To make sure the structure is maintained, the first term $(M_P - M'_P)$ is invoked to make the final illumination map as identical to the initial illumination map. λ is a weight assigned for balancing the two terms.

This improved and refined illumination map allows to recover the enhanced version I' , which is the desired exposure corrected image as follows:

$$I' = I \times M^{-1} \quad (5)$$

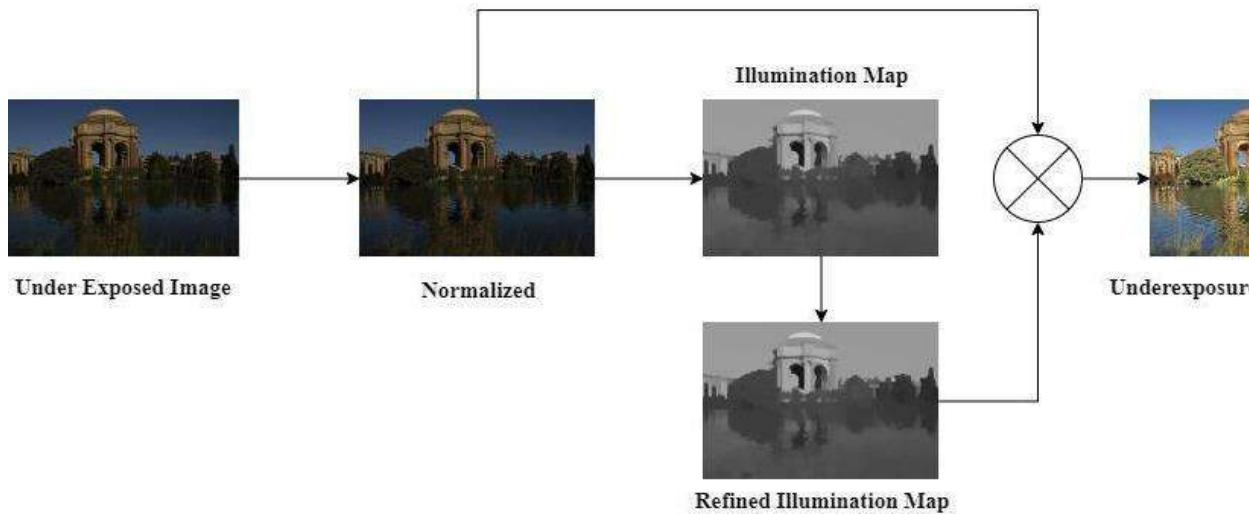


Fig 2. Exposure correction mechanism on underexposed images.

The aforementioned steps are strictly followed for the underexposed images, as represented in Fig. 2. For overexposed images, their inverted image is taken for the above mentioned procedure. After acquiring the enhanced inverted image, the overexposed image's exposure is rectified by flipping the image back to its normal format. Once this is done, the exposure-corrected variants of both the underexposed and overexposed images are obtained, namely, underexposed corrected and overexposed corrected images.

Although these developed images after exposure correction are noticeably sharper and more detailed, the overall authenticity of the natural scene will be jeopardized by the excessive brightness. The proposed solution is a traditional way for keeping edges and texture information using edge-preserving filters[18,19,20]. Among them, an edge-preserving recursive filter is used to create an intermediate refined exposure stack with fine edges and less noise by taking advantage of both the input and exposure-corrected images. The key idea is to ensure that the pixels in the images fed onto the filter have comparative weights. The recursive filter [21] provides much more visually pleasing results and claims to have higher computational efficiency with a promising success rate among multi-exposure fusion techniques.

A recursive edge-preserving [22] filter is executed as follows:

$$R[i] = (1 - c^d)I[i] + c^d R[i-1] \quad (6)$$

where c is the feedback coefficient $[0,1]$ and d represents the distance between pixels. As the distance increases the coefficient goes to 0. $I[i]$ is the value of the i^{th} pixel of the input image, and $R[i]$ is the i^{th} pixel of the refined image.

3.2 Pyramidal Decomposition on the Refined Exposure Stack

The best elements of the refined images are combined to create a globally well-exposed image. The proposed method fuses the intermediate results obtained after the edge-preserving refinement for obtaining the resultant image. In addition to a set of quality metrics, weighted blending is used in the procedure. The idea is to assign fewer weights to the flat colorless regions and more weights to the richer regions.

Let I_{UR} and I_{OR} be the underexposed refined and overexposed refined images obtained. For the intermediate refined exposure stack $\{I_{UR}, I_{OR}\}$, a weight map is generated using the qualitative measures of contrast, saturation, and well-exposedness [5].

Contrast measurement aids in identifying the input image's textures and edges. The value is acquired by taking the absolute value of the laplacian filter response of the image's grayscale variant [23] as follows:

$$C(x,y) = |I_{gray}(x,y) * l(x,y)| \quad (7)$$

where, I_{gray} stands for the grayscale version of refined images, $*$ is a convolution operator, C is for the contrast, and l points to the Laplacian filter.

The saturation gives the image's colors a more vibrant appearance, and a vivid image is crucial. The colors produced by lengthy exposures will ultimately clip and become desaturated. The standard deviation of the RGB color space is used to compute the saturation value S , as follows:

$$S(x,y) = \sqrt{\frac{(R - \mu)^2 + (G - \mu)^2 + (B - \mu)^2}{3}} \quad (8)$$

where,

$$\mu = \frac{R + G + B}{3} \quad (9)$$

Well-exposedness aids in removing overexposed and underexposed areas from the source images. The measure is calculated by taking the gaussian curve of each channel and selecting the pixels that have values close to 0.5 as follows:

$$E(x,y) = R_g + G_g + B_g \quad (10)$$

where, R_g , G_g , and B_g are the Gaussian curves of the red, green, and blue channel of the refined images respectively and which are obtained by

$$\exp\left(-\frac{(i - 0.5)^2}{2\sigma^2}\right) \quad (11)$$

Here the value of σ is 0.2. For each exposure-corrected image, a single refined map is created after characterizing all three weight maps as follows:

$$W(x,y) = C(x,y) \times S(x,y) \times E(x,y) \quad (12)$$

A pyramidal decomposition is now used to merge the refined exposures, ensuring that the final image is devoid of artifacts at color shifts and has sharp texturing [3]. Here, the corrected exposure stack is divided into ℓ levels of different resolutions using the Laplacian pyramid (L). On fused weights, a gaussian pyramid (G) is utilized for the similar process. Each level of these pyramids undergoes the blending operation, resulting in the formation of a laplacian pyramid from the combined image, as follows:

$$L\{f^l\} = \sum_{n=1}^N G\{W_n^l\} \times L\{I_n^l\} \quad (13)$$

where W represents the fused weight and I represents the refined image. In order to obtain the final fused image F , the fused pyramid $L\{f^l\}$ is eventually collapsed.

4 Results and Analysis

The performance of the proposed approach is assessed both qualitatively and quantitatively by conducting various experiments on test images taken from datasets in [2].

Table 1. Quantitative methods of different MEF algorithms using PSNR. The best three values in the order are shown by the colours red, green, and blue.

	Mertens09	Liu15	Lee18	Hayat19	Li20	Fang20	Wang20	Karakaya22	Proposed method
<i>Arno</i>	55.7303	55.5498	55.7403	55.7155	55.6729	55.7503	55.6731	55.743	55.7459
<i>House</i>	54.0031	53.7052	53.8625	53.836	53.6219	53.9169	53.6865	53.8713	53.99
<i>Kluki</i>	55.379	55.0229	55.6258	55.3423	55.3096	55.3417	55.1474	55.4054	55.4238
<i>Landscape</i>	56.2406	55.8549	56.2722	56.2255	56.2744	56.259	55.9825	56.2675	56.3206
<i>Lighthouse</i>	55.1125	54.793	54.9861	55.1118	54.9586	55.0924	54.7418	55.0702	55.1664
<i>Venice</i>	55.7405	55.4963	55.8662	55.7273	55.6711	55.6628	55.3307	55.6695	55.8288
<i>AirBellowsGap</i>	55.3297	55.0839	54.9355	55.2971	55.0443	55.2971	54.8165	55.7227	55.3547
<i>BarHarborSunrise</i>	55.552	55.1161	55.5395	55.4729	55.3549	55.6172	55.4636	55.6621	55.5572
<i>BloomingGorse</i>	54.8485	54.3895	54.7604	54.7994	54.5843	54.8361	54.5414	55.0464	54.9006
<i>Exploratorium</i>	54.8103	54.477	54.6486	54.7342	54.6165	54.7772	54.3793	54.5398	54.8544
<i>Knossos6</i>	55.2642	54.9017	55.1868	55.2035	55.1434	55.1977	54.9694	55.1646	55.2279
<i>Museum1</i>	54.3245	54.0713	54.3208	54.2773	54.3133	54.3215	54.2779	54.34	54.3275
<i>Stream</i>	55.6513	55.4341	55.5132	55.5816	55.3485	55.6259	55.3036	56.0038	55.6422
<i>Chair</i>	53.6458	53.6949	53.7172	53.6255	53.1284	53.7656	53.6869	53.6486	53.7627

<i>WindowTrim</i>	55.9436	55.62	55.9125	55.8081	55.7469	55.8419	55.7361	55.898	55.9512
<i>Average</i>	55.1717	54.8807	55.1258	55.1172	54.9859	55.1536	54.9158	55.2035	55.2036

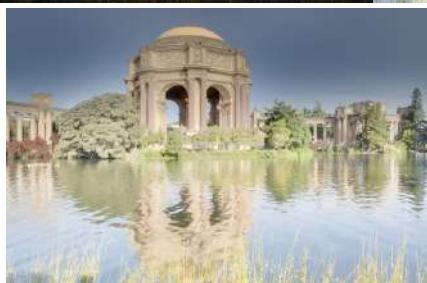
The proposed method's outcomes are compared with 8 existing methods which include, Mertens [5], Liu [6], Lee [7], Hayat [8], Li [9], Fang [10], Wang [11], and Karakaya [12]. The experiments were performed in a Core i9-equipped CPU clocked at 3.70 GHz supported by 32 GB RAM. All fused images are generated under default settings without any optimizations with the publicly available implementations. Statistical analysis is performed using a Structural Similarity Index Metric for MEF (MEF-SSIM) [24] and Peak Signal-to-Noise Ratio (PSNR)[25] over 15 test images. MEF-SSIM is a structural similarity-based metric that generates results by taking brightness, global consistency, and structure preservation into account. An outcome closer to 1 denotes a greater structural similarity. PSNR is a commonly used information theory-based metric, calculated by dividing the number of gray levels in the image with the corresponding pixels in fused and reference images. A higher PSNR value denotes a stronger relationship between the reference and fused images, denoting a superior fusion.

The results of the statistical analysis using, PSNR, and MEF-SSIM, are shown in Tables 1 and 2, respectively. The average accuracy is shown in the bottom rows of both tables, and it is clear that the given strategy performs better than the compared state-of-the-art MEF approaches.

Table 2. Quantitative methods of different MEF algorithms using MEF-SSIM. The top three results for the specified measure are shown in red, green, and blue, accordingly.

	Mertens09	Liu15	Lee18	Hayat19	Li20	Fang20	Wang20	Karakaya22	Proposed method
<i>Arno</i>	0.9878	0.943	0.9818	0.9849	0.9798	0.9857	0.9748	0.9735	0.9882
<i>House</i>	0.9723	0.9429	0.9654	0.9706	0.9541	0.9725	0.9458	0.9601	0.9667
<i>Kluki</i>	0.9801	0.9605	0.965	0.9849	0.98	0.9844	0.9668	0.9742	0.978
<i>Landscape</i>	0.9734	0.9493	0.971	0.9726	0.9743	0.9735	0.9531	0.974	0.9782
<i>Lighthouse</i>	0.9754	0.9499	0.9687	0.9777	0.9372	0.974	0.956	0.9749	0.9758

<i>Venice</i>	0.9818	0.9495	0.9747	0.9825	0.927	0.9868	0.9632	0.9843	0.9828
<i>AirBellowsGap</i>	0.9673	0.9447	0.9507	0.9658	0.9429	0.9644	0.9405	0.9461	0.97
<i>BarHarborSunrise</i>	0.9747	0.9471	0.9639	0.9763	0.9637	0.9824	0.9429	0.8998	0.9771
<i>BloomingGorse</i>	0.9763	0.9022	0.9638	0.958	0.9513	0.9717	0.9183	0.9623	0.9795
<i>Exploratorium</i>	0.9687	0.9146	0.9664	0.9617	0.8843	0.967	0.9608	0.972	0.9737
<i>Knossos6</i>	0.9734	0.9156	0.9609	0.9708	0.9663	0.974	0.9313	0.9635	0.974
<i>Museum1</i>	0.9933	0.9494	0.988	0.9882	0.9891	0.9928	0.9636	0.9891	0.9934
<i>Stream</i>	0.9755	0.9262	0.95	0.9759	0.9004	0.9749	0.9207	0.9326	0.9758
<i>Chair</i>	0.9472	0.9301	0.95	0.9431	0.6028	0.9319	0.9172	0.9372	0.9474
<i>WindowTrim</i>	0.9673	0.9173	0.9557	0.9638	0.9346	0.9695	0.9202	0.9424	0.9624
<i>Average</i>	0.9743	0.9362	0.9651	0.9718	0.9259	0.9737	0.945	0.9591	0.9749



(a) Underexposed

(b) Overexposed

(c) Wang



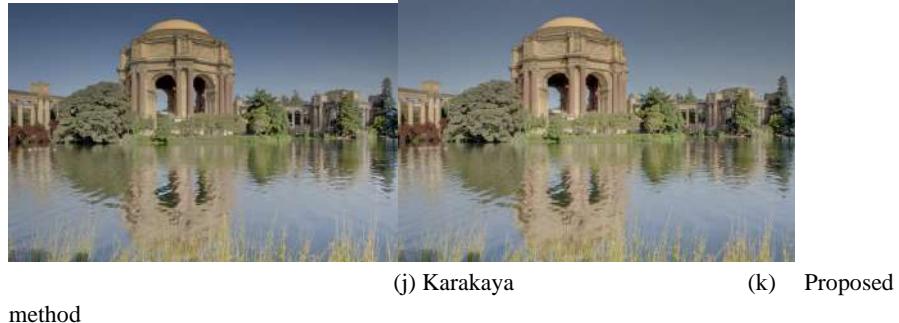
(d) Liu
(f) Hayat

(e) Li



(g) Lee
(i) Fang

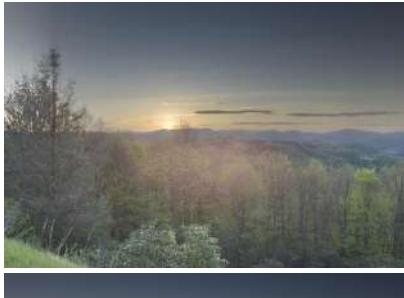
(h) Mertens



method

Fig. 3. The qualitative performance comparison using the ‘Exploratorium’ image sequence





(d) Liu
(f) Hayat

(e) Li



(g) Lee
(i) Fang

(h) Mertens



method

Fig. 4. The qualitative performance comparison using ‘AirBellowsGap’ image sequence. Figure 3 shows the result of different MEF algorithms including the proposed method on ‘Exploratorium’ image sequence. In the proposed method, the transition between the buildings and the sky component is smoothly blended with improved contrast, but for the others, this connected area is distorted. The textural elements of the building were lost in the outcomes depicted by Fang and Wang. Liu’s fused image shows relatively few edge details at the junction between the river and the building. The overall quality of the image suffers as a result. The result shown by Hayat also shows similar detail loss at the building part. Lee’s fused image has brighter sections that are prone to losing detail, whereas, Karakaya provided outcomes that were slightly darker.

Figure 4 shows the results of different MEF algorithms including the proposed work on ‘AirBellowsGap’ image sequence. The proposed method exhibits standard visual quality, with strong color saturation and texture information in the tree section. The sky appears to be excessively bright and has lost details in the resultant image of Karakaya. Fang and Mertens display the fused image with comparable contrast and sufficient edge details. Wang and Liu’s fused images seem to fade out with lower quality. From figures 3 and 4, it is evident that the proposed method outperforms the existing methods.

5 Conclusion

MEF is frequently used to create high-quality, HDR like images from a collection of LDR images. In the proposed method, the input image is enhanced using an exposure-correction approach before going through the pyramidal fusion. This technique enhances the quality of input images in order to produce a fused image with improved edge, texture, and contrast figures. Initially, the input images are

refined using a recursive filter with the exposure-corrected variants of the input images. Image fusion on these refined images is carried out using a pyramidal decomposition. The experimental comparisons reveal the superiority of the proposed method over a number of state-of-the-art MEF methods. In future, the performance of the fusion can further be improved by setting the proposed strategy into practice with the aid of deep learning. The goal is to focus on particular areas of an image rather than assuming that an entire image qualifies as being underexposed or overexposed. This will hopefully give more pleasing results and higher accuracy.

References

1. Reinhard, E., Heidrich, W., Debevec, P., Pattanaik, S., Ward, G., & Myszkowski, K. (2010). High dynamic range imaging: acquisition, display, and image-based lighting. Morgan Kaufmann.
2. Zhang, X. (2021). Benchmarking and comparing multi-exposure image fusion algorithms. *Information Fusion*, 74, 111-131.
3. Burt, P. J., & Kolczynski, R. J. (1993, May). Enhanced image capture through fusion. In 1993 (4th) international Conference on Computer Vision (pp. 173-182). IEEE.
4. Goshtasby, A. A. (2005). Fusion of multi-exposure images. *Image and Vision Computing*, 23(6), 611-618.
5. Mertens, T., Kautz, J., & Van Reeth, F. (2007, October). Exposure fusion. In 15th Pacific Conference on Computer Graphics and Applications (PG'07) (pp. 382-390). IEEE.
6. Liu, Y., & Wang, Z. (2015). Dense SIFT for ghost-free multi-exposure fusion. *Journal of Visual Communication and Image Representation*, 31, 208-224.
7. Lee, S. H., Park, J. S., & Cho, N. I. (2018, October). A multi-exposure image fusion based on the adaptive weights reflecting the relative pixel intensity and global gradient. In 2018 25th IEEE International Conference on Image Processing (ICIP) (pp. 1737-1741). IEEE.
8. Hayat, N., & Imran, M. (2019). Ghost-free multi exposure image fusion technique using dense SIFT descriptor and guided filter. *Journal of Visual Communication and Image Representation*, 62, 295-308.
9. Li, H., Ma, K., Yong, H., & Zhang, L. (2020). Fast multi-scale structural patch decomposition for multi-exposure image fusion. *IEEE Transactions on Image Processing*, 29, 5805-5816.
10. Fang, Y., Zhu, H., Ma, K., Wang, Z., & Li, S. (2019). Perceptual evaluation for multi-exposure image fusion of dynamic scenes. *IEEE Transactions on Image Processing*, 29, 1127-1138.
11. Wang, Q., Chen, W., Wu, X., & Li, Z. (2019). Detail-enhanced multi-scale exposure fusion in YUV color space. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(8), 2418-2429.
12. Karakaya, D., Ulucan, O., & Turkan, M. (2022, May). PAS-MEF: Multi-exposure image fusion based on principal component analysis, adaptive well-exposedness and saliency map. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 2345-2349). IEEE.
13. Hanmandlu, M., Verma, O. P., Kumar, N. K., & Kulkarni, M. (2009). A novel optimal fuzzy system for color image enhancement using bacterial foraging. *IEEE Transactions on*

- Instrumentation and Measurement, 58(8), 2867-2879.
- 14. Wang, S., Zheng, J., Hu, H. M., & Li, B. (2013). Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE transactions on image processing*, 22(9), 3538-3548.
 - 15. Guo, X., Li, Y., & Ling, H. (2016). LIME: Low-light image enhancement via illumination map estimation. *IEEE Transactions on image processing*, 26(2), 982-993.
 - 16. Zhang, Q., Yuan, G., Xiao, C., Zhu, L., & Zheng, W. S. (2018, October). High-quality exposure correction of underexposed photos. In *Proceedings of the 26th ACM international conference on Multimedia* (pp. 582-590).
 - 17. Zhang, Q., Nie, Y., & Zheng, W. S. (2019, October). Dual illumination estimation for robust exposure correction. In *Computer Graphics Forum* (Vol. 38, No. 7, pp. 243-252).
 - 18. Biradar, N., Dewal, M. L., & Rohit, M. K. (2014). Edge preserved speckle noise reduction using integrated fuzzy filters. *International scholarly research notices*, 2014.
 - 19. Biradar, N., Dewal, M. L., & Rohit, M. K. (2014). Edge preserved speckle noise reduction using integrated fuzzy filters. *International scholarly research notices*, 2014.
 - 20. Jain, P., & Tyagi, V. (2016). A survey of edge-preserving image denoising methods. *Information Systems Frontiers*, 18(1), 159-170.
 - 21. Li, S., & Kang, X. (2012). Fast multi-exposure image fusion with median filter and recursive filter. *IEEE Transactions on Consumer Electronics*, 58(2), 626-632.
 - 22. Gastal, E. S., & Oliveira, M. M. (2011). Domain transform for edge-aware image and video processing. In *ACM SIGGRAPH 2011 papers* (pp. 1-12).
 - 23. Malik, J., & Perona, P. (1990). Preattentive texture discrimination with early vision mechanisms. *JOSA A*, 7(5), 923-932.
 - 24. Ma, K., Zeng, K., & Wang, Z. (2015). Perceptual quality assessment for multi-exposure image fusion. *IEEE Transactions on Image Processing*, 24(11), 3345-3356.
 - 25. Jagalingam, P., & Hegde, A. V. (2015). A review of quality metrics for fused image. *Aquatic Procedia*, 4, 133-142.

Resume Analysis Using NLP

*Rakhi Bhardwaj¹, Divya Mahajan², Meenal Bharsakle³, Kashish Meshram⁴, and
Himaja Pujari⁵

Vishwakarma Institute of Technology, Pune

[1rakhi.bhardwaj@vit.edu](mailto:rakhi.bhardwaj@vit.edu)

[2divya.mahajan21@vit.edu](mailto:divya.mahajan21@vit.edu)

[3meenal.bharsakle21@vit.edu](mailto:meenal.bharsakle21@vit.edu)

[4kashish.meshram21@vit.edu](mailto:kashish.meshram21@vit.edu)

[5himaja.pujari21@vit.edu](mailto:himaja.pujari21@vit.edu)

Abstract. Job recruitment is one of the main activities for individuals, and it might be difficult to discover a fruitful talent. Our suggested model's main elements include the extraction of statistics and information from the resume and the evaluation of the resume in line with the preferences of the associated firm and its criteria for natural language processing (NLP). The hiring process is simplified and made more effective by parsing and ranking the resume. Any reputable parser must be able to extract the different tiny pieces of information contained in a resume, including information on education, experience, projects, addresses, and other things. So, in essence, we'll create a job platform where employees and candidates may upload their resumes for every available position. The required information will be parsed with NLP. Additionally, the corporate skill requirements and the talents of the employees in the provided CV will be taken into consideration when ranking the employee resumes.

Keywords: Natural Language Processing, parser, ranking, resumes, skillset.

1 Introduction

Many resumes are processed everyday by corporate organizations and recruitment firms. This task is not easy for humans. It takes an automated, intelligent system to extract all the relevant data from resumes and ranked for a position. Many individuals send their resumes to apply for a particular position, and the multinational businesses receive thousands of emails from them each day. The actual difficulty now is determining which resumes should be sorted and shortlisted based on the requirements. The resume can be manually examined and sorted as one approach.

Due to human participation, this method is currently the most time-consuming and has a high potential for errors. Additionally, people cannot work continuously. Consequently, there is also a problem with efficiency.

The following information is parsed: name, email address, social media accounts, personal websites, years of employment, employment experiences, years of education, educational experiences, publications, certificates, volunteer experiences, keywords, and finally the cluster of the resume. Discrete sets are used to store data. Each set includes information on a person's contact information, job history, and educational background. Despite this, it can be challenging to parse resumes. This is due to the fact that they have different informational types, orderings, writing styles, etc. They can be written in a number of different formats. They include ".txt", and ".pdf" and are the most often used ones. In order to correctly and effectively understand the data from various types of resumes, the model must not be dependent on the order or type of data.

2 Literature Review

The research articles that we have read and examined demonstrate the necessity for both candidates and recruiters to review resumes. Sufficient database is significant for the model. Accuracy of the model is always a concern. Also, skillsets and job description play a vital role in resume analysis. Keeping the user interface of the website simple, would be encouraging more users to visit the website.

The work by Shubham Bhor, et al. depicted the use of NLP for parsing the resume according to particular company [1].

The system proposed by Sroison Pornphat and Jonathan H. consisted of three steps to parse resume; firstly, receive resume files from candidate, then convert resume file to text format, and finally extracting necessary information [2].

Yi-Chi Chou and Han-Yen Yu have proposed an AI model which analyses resume of different candidates and gives them a score based on the skills that are mentioned [3].

A methodological approach for processing and analyzing data collected by scraping from online job portals using AI is proposed by Alena Vankevich and Iryna Kalinouskaya [4].

The system by V. V. Dixit, et al. uses AI to find required skillset by scanning resume and also sorts it [5].

The proposed CV parser by Papiya Das, et al., system provides entity extraction method from the uploaded CV's [6].

The system by Yanyuan Su Jian and Zhang Jianhao Lu performs a series of experiments to train and estimate several neural network models on it [7].

This system proposed by Ashif Mohamed, et al., uses NLP and pattern matching technique to extract details of the candidate [8].

Paper Title	Authors	Future Scope
Resume Parser Using Natural Language Processing Techniques (2021)	Shubham Bhor, Vivek Gupta, Vishak Nair, Harish Shinde	To parse resumes from different websites and applications like LinkedIn, GitHub, Naukri.com etc.
Resume Parser with Natural Language Processing (2021)	Sroison Pornphat, Jonathan H.	A system that provides more datasets for training in the future as the existing datasets are inadequate.
Based on the application of AI technology in resume analysis and job recommendation (2020)	Yi-Chi Chou, Han-Yen Yu	The accuracy of the system can be enhanced, this study only examined the feedback of applicant who used the job vacancy recommendation systems.
Ensuring sustainable growth based on the artificial intelligence analysis and forecast of in demand skills (2020)	Alena Vankevich, Iryna Kalinuskaya	To identify the skills and competencies in demand and compare them with those offered on the labour market.
Resume Sorting using Artificial Intelligence (2019)	V. V. Dixit, Trisha Patel, Nidhi Deshpande, Kamini Sonawane	To use augmented intelligence technology to determine applicant's culture fit and improve relationship with hiring executives.
A CV Parser Model using Entity Extraction Process and Big Data Tools (2018)	Papiya Das, Manjusha Pandey, Siddharth Rautray	To implement and deliver a brief analysis in real time database to attain the analysis with present models.
The Resume Corpus: A large Dataset for Research in information Extraction System (2018)	Yanyuan Su Jian, Zhang Jianhao Lu	To design a series of information extraction models based on the corpus.
Smart Talents Recruiters - Resume Ranking and Recommendation System (2018)	Ashif Mohamed, Shahik Samarth, Anuradha Jayakody, Usama Iqbal	To examine important data about job recruiting and extract more information to implement enriched applicant recommendation system and enlarge the database.

Table. 1. Summary of Literature Review

Major problems faced in previous works is the unavailability of proper database and they couldn't figure out which parameters should be used to compare and rank the resumes.

3 Methodology

3.1 Flowchart

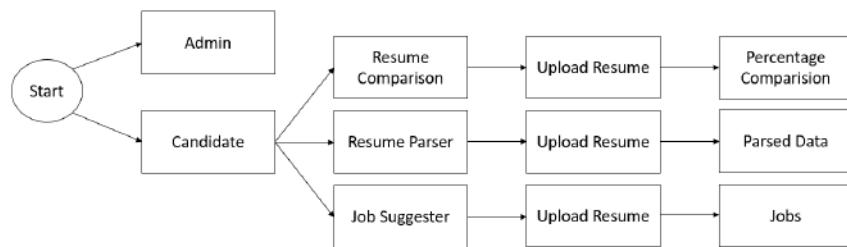


Fig. 1. System Architecture

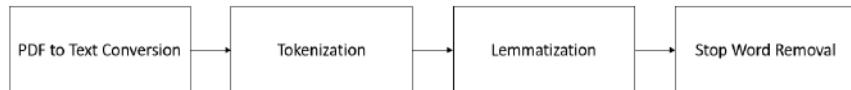


Fig. 2. Pre-processing steps

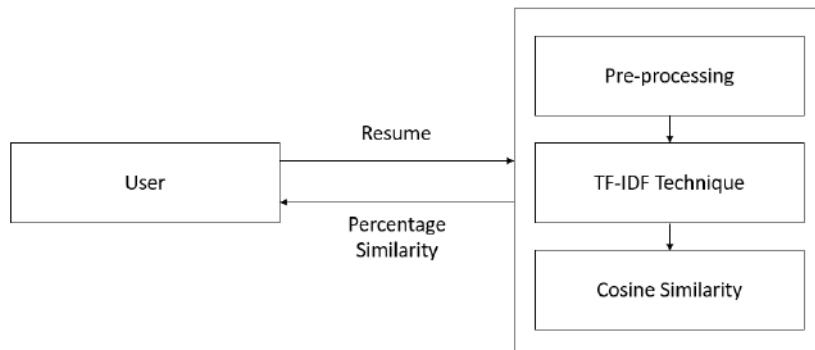


Fig. 3. Generating Response steps

3.2 Components

An interface between the recruiter and the candidates may be found on the site. A completely working website is made using Flask, HTML, CSS, JavaScript, and other tools. Making the website user-friendly is the aim. It may be used by applicants and recruiters for many purposes. The website is quite easy to navigate.

Flask is a web framework developed in Python that is used to offer a backend to a website and add functionality to it. We wrote our AI model in Python, and using Flask made it easy for us to put the code on a website.

3.2 Algorithm

In this study, natural language processing (NLP) is employed. NLP enables computers to comprehend human language and process it accordingly. The Natural Language Toolkit (nltk) library is utilized for this. Other libraries are imported and installed in addition to nltk. It is a good library when a specific combination of algorithms is required.

A web application with two ends; the applicant and the recruiter, is where the model is deployed. NLP techniques are significantly portrayed at each finish. The modules that follow will demonstrate this.

The applicant module:

- Resume Parser

Utilizing the Python module pdfminer, which is used in NLP, the text from the resume is extracted. The resume parser extracts email IDs, phone numbers, skills, and education using Regular Expressions (RE) and Name Entity Recognition (NER).

1. Email IDs

Email IDs are collected from the resume using pattern matching and the following regular expression. The re library is utilized for this operation.

$$RE = ([^@|\s] + @[^@] + \. [^@|\s]+)$$

2. Phone number

Similar to email IDs, the phone number is obtained using another regular expression.

3. Skills

It makes use of a dataset that is in .csv format and comprises a variety of skills. The column values in the dataset are compared to the tokens in resume. A match clearly shows that the skill is present in the resume if one is identified. Consequently, the talents can be taken from the CV.

4. Education

The text from the resume is compared with the list of educational qualifications and thus, if a match is found, it is returned.

- **Resume Evaluator**

The candidate can select the job category with which they want to compare and upload their CV. The resume analyzer will review the document and report the percentage of similarity between it and the selected category's job description.

Before the real procedure is carried out, the resume needs to go through some pre-processing.

1. Pre-processing

The information collected from the user is pre-processed using a variety of NLP algorithms. The first step is using the pdfminer library to convert the resume to plain text. The following techniques are used to process it after that.

- a. Tokenization – Data is divided into chunks by the tokenizer so that they can be handled as discrete parts in subsequent phases. It uses the Multi-Word Expression Tokenizer (MWETokenizer).
- b. Lemmatization - The terms are condensed by lemmatizer to their simplest versions. The Wordnet lemmatizer is employed in this work.
- c. Stop word removal – It eliminates all the unnecessary and frequent words. In simple words, it helps get rid of all the words that are unnecessary and overused.

2. Generating Response

The only thing left over when pre-processing is completed is the clean text. The following procedures are then used to process this further.

- a. The TF-IDF methodology - It counts the frequency of each character and divides it by the total number of words, is used to determine the TF (Term Frequency) of each word. The IDF (Inverse Document Frequency) of the vector is then calculated using the following formula, and the two terms are multiplied. With the use of this technology, each word in the text is transformed into a vector. Both the CV and the job description go through this process. TF-IDF vectorization is used in the model since

it if reduces the importance of the common words unlike bag of words approach which simply counts the number of occurrences.

$$TF(t, d) = \frac{\text{Number of occurrences of term } t \text{ in data } d}{\text{Total number of characters/terms in data } d}$$

$$IDF(t) = \log \frac{\text{Total number of documents}}{1 + \text{Number of documents containing the term } t}$$

$$TF-IDF(t, d) = TF(t, d) * IDF(t)$$

- b. Cosine Similarity – Utilizing cosine similarity, the vectors derived from the job description and resume are compared.

$$\text{Cosine Similarity}(d1, d2) = \frac{\text{Dot Product}(d1, d2)}{\|d1\| * \|d2\|}$$

where there are two non-zero vectors, d1 and d2.

- Job Suggester

We scrapped a website named Glassdoor to obtain information about various job opportunities that are open to candidates. Web scraping is a method of collecting data from webpages and converting it into a structured format. Scrapy and beautiful soup are Python libraries that are used for web scraping.

1. N-gram

Continuous word, symbol, or token sequences in a document are known as n-grams. Technically speaking, they might be referred to as the adjacent item sequences in a document. The entire text is divided into clusters of size n by this approach. Through the TF-IDF approach, these clusters are tokenized and then further vectorized. This makes it easier to understand the context in which a term is used.

2. K Nearest Neighbor

The k nearest neighbor algorithm, or KNN for short, is a supervised learning classifier that relies on closeness to generate classifications or forecasts for how a certain data point will be categorized. In this study, KNN determines the closeness between the job description vector and the vector produced by the N-gram approach, resulting in a similarity between both.

The Admin Module:

The compared resumes are stored in the database. These resumes are displayed on the Admin portal.

4 Results and Discussion

The model is successful in analyzing the resumes and ranking them according to the job description provided by the recruiter. The education of the applicant, experience, projects covered, etc. are considered during the analysis. Other important details such as name of the applicant, email ID, skillset, etc. are extracted from the resume. In comparison to other models, our model is useful for both the recruiter and candidates as it compares the resume for the recruiters and gives job suggestions to the candidates based on their skills. Following figures illustrate the code and resulting output of the project.

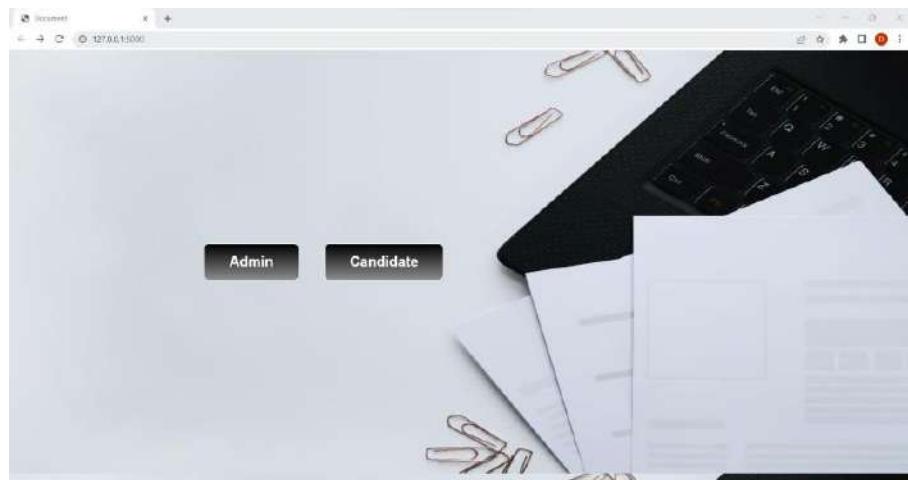


Fig. 4. Home Page



Fig. 5. Candidate Portal



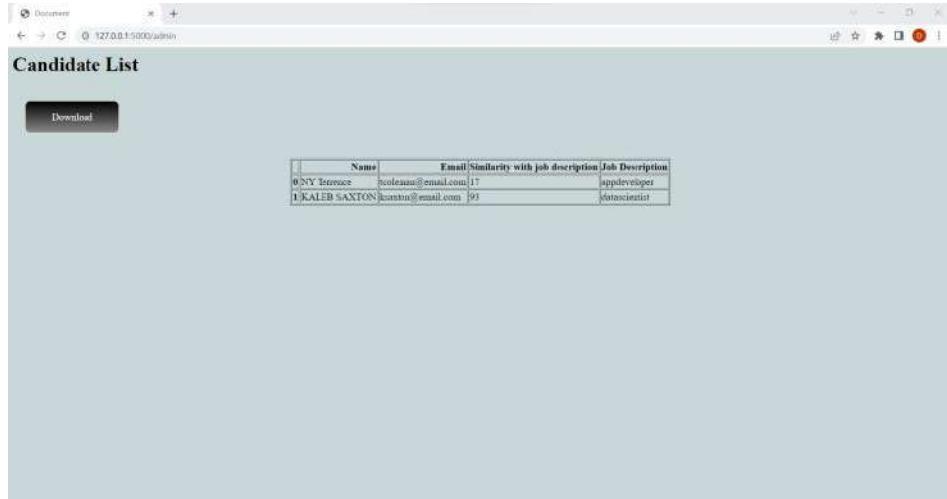
Fig. 6. Resume Comparison

The screenshot shows a web browser window titled "Document" with the URL "127.0.0.1:5000/resume_parser". The page contains a form for uploading a resume. At the top, there is a file input field labeled "Choose File" with the placeholder "No file chosen" and a "Upload" button. Below the file input, there are five text input fields with labels: "Name:", "Mobile number:", "Email:", "Education:", and "skills:". The entire form is set against a light gray background.

Fig. 7. Resume Parser

The screenshot shows a web browser window titled "Home page" with the URL "127.0.0.1:5000/v1/jobs.sugg". The page has a light blue background. At the top, it displays the text "Upload your resume" above a file input field labeled "Choose File" with the placeholder "No file chosen" and a "Submit" button. Below this, the text "Suggested jobs" is displayed in a larger font. The rest of the page is blank, showing a large white area.

Fig. 8. Job Suggestor



The screenshot shows a web browser window titled 'Document' with the URL '127.0.0.1:5000/admin'. The main content area is titled 'Candidate List' and contains a table with two rows of data. A 'Download' button is located above the table.

	Name	Email	Similarity with job description	Job Description
0	NY Terence	coleann@email.com	17	appdeveloper
1	KALEB SAXTON	konstan@email.com	93	datascientist

Fig. 9. Admin Portal

5 Future Scope

Our project's primary area of future work will be to parse resumes from many applications and websites, like LinkedIn, GitHub, Naukri.com, etc. This system can be expanded in the future to include a wide variety of psychometric exams. Future work will involve expanding the resume collection and enhancing the functionality of the suggested system.

6 Conclusion

Each day, countless resumes are managed by different corporate organizations and job agencies. A computerized intelligence system ranks the resumes based on the pertinent data that is taken from them and ranked for a particular job opportunity. The data that was studied comprises name, email address, social media accounts, personal websites, years of work, employment experiences, years of education, educational experiences, publications, certifications, volunteer experiences, and keywords. The employment process's simplification is our primary objective. Businesses will receive eligible applications as a result of this process. There will be fewer of the unfair and discriminatory behaviors throughout the process.

7 Acknowledgement

We are extremely grateful to the project guide, Prof. Rakhi Bharadwaj, who guided us well in the completion of our project. We are also thankful for the opportunity provided by the Head of Computer Science Department, Prof. Dr. Sandip Shinde. This project would not have been possible without their guidance. Because of the moral support and ideas, this project emerged successful.

References

1. Shubham Bhor, Vivek Gupta, Vishak Nair, Harish Shinde. "Resume Parser Using Natural Language Processing Techniques". International Journal of Research in Engineering and Science (IJRES). Vol 9, 24 June 2021.
2. Sroison ,Pornphat ,Jonathan H." Resume Parser with Natural Language Processing " 2021.
3. Yi-Chi Chou, Han-Yen Yu "Based on the application of AI technology in resume analysis and job recommendation", 2020. IEEE International Conference on Computational Electromagnetics (ICCEM 2020).
4. Alena Vankevich, Iryna Kalinowskaya "Ensuring sustainable growth based on the artificial intelligence analysis and forecast of in demand skills ",2020. E3S Web of Conferences 208.
5. V. V. Dixit, Trisha Patel, Nidhi Deshpande, Kamini Sonawane "Resume Sorting using Artificial Intelligence", 2019. International Journal of Research in Engineering, Science and Management Volume-2, Issue-4, April-2019.
6. Papiya Das, Manjusha Pandey, Siddharth Rautaray "A CV Parser Model using Entity Extraction Process and Big Data Tools",2018 I.J. Information Technology and Computer Science, 2018.
7. Yanyuan Su Jian, Zhang Jianhao Lu "The Resume Corpus: A large Dataset for Research in information Extraction System" (2018). 15th International Conference on Computational Intelligence and Security (CIS) 2018.
8. Ashif Mohamed, Shahik Samarth, Anuradha Jayakody, Usama Iqbal "Smart Talents Recruiters - Resume Ranking and Recommendation System" 2018. IEEE International Conference on Information and Automation for Sustainability (ICIAfS) 2018.
9. Bhardwaj, R. J., & Rao, D. (2022). Deep Learning-Based Traffic Behavior Analysis under Multiple Camera Environment. International Journal of Next-Generation Computing.
10. N. Pavithra, P. Bhatele, S. Desai and H. Pande, "Design and Implementation of Multipurpose Chatbot," 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), 2022, pp. 1332-1337, doi: 10.1109/ICSSIT53264.2022.9716306.
11. Tiedan Zhu, Kan Li, The Similarity Measure Based on LDA for Automatic Summarization, Procedia Engineering, Volume 29, 2012, Pages 2944-2949, ISSN 1877-7058.
12. Ryu, Keun Ho. "BioBERT Based Efficient Clustering Framework for Biomedical Document Analysis." Genetic and Evolutionary Computing: Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computing, October 21-23, 2021, Jilin, China. Vol. 833. Springer Nature, 2022.

Estimation of Queuing System Incoming Flow Intensity

*Alexander Zyulkov¹, Yury Kutoyants^{2,3,4}, Svyatoslav Perelevsky⁵, and Larisa Korableva⁶

¹Voronezh State University, Voronezh, Russia

^{1*}a.v.zyulkov@mail.ru

²Le Mans University, Le Mans, France

³National Research University “MPEI”, Moscow, Russia

⁴National Research Tomsk State University, Tomsk, Russia

^{2,3,4}youri.koutoyants@univ-lemans.fr

⁵National Research Tomsk State University, Tomsk, Russia

⁵s.perelevskiy@mail.ru

⁶Moscow State University, Moscow, Russia

⁶mail@mse-msu.ru

* Corresponding Author

Abstract. Based on the proposed simple probabilistic model of the utilization factor of a single-channel queuing system, a maximum likelihood estimate of the intensity of the incoming flow of requests and its statistical characteristics are found. It is shown that the synthesized estimate can be explicitly represented analytically and allows a technically much simpler implementation compared to its common analogues, with comparable or better accuracy and shorter observation time intervals. The obtained theoretical results are confirmed experimentally during simulation by means of AnyLogic software.

Keywords: Queuing system, flow intensity, utilization factor, probabilistic model, probabilistic mixture, maximum likelihood estimate, AnyLogic environment, reinforcement learning.

1 Introduction

Evaluation of the performance of the computer systems is mandatory for the developers whose task is to determine the optimal configuration at the system design phase when the number of CPUs and the memory size are chosen. For this purpose, as models, both queuing systems and networks are usually used allowing quantifying system performance by simulating stochastic dynamic behavior. It means simulating the requests of incoming and servicing processes. Their intensity must be set based on the observation of the other system characteristics.

In [1], one observes a series of incoming and outgoing requests at a fixed time

with a subsequent estimation of the arrival rate and service rate in a simple queue. In [2], the estimation of the service time and waiting time in the GI|G|1 system is carried out based on the data on the waiting time in the queue. In [3], the focus is on the analysis of the estimate of the arrival and service rates. It is performed on the data on the queue length at successive time periods in an M|M|K system. In [4], one considers the estimates of the arrival and service rates in the M|M|1 system. In [5], one deals with the determining the arrival rate and service rate in the M|M|1 queuing system based on the data on the time of request arrival and the time of its service. To accomplish this task, an empirical Bayesian approach is applied. In [6], the intensity of traffic is estimated based on the data on the M|M|1 system load.

In [2], [3], [7], it is assumed that the time of arrival and the number of arrivals in the time interval are observable meaning that the exact job arrival time and service time are known. However, the problem with the evaluation of the computer system arrival processes is that the observations that can be used for evaluation are very limited.

Among the observable system characteristic of systems, there is utilization data, defined as time series data consisting of time fractions of busy periods in the fixed time intervals. In practice, they are used to represent server conditions such as CPU usage. The definition of utilization data shows that the data collection process is carried out only in a series of discrete time intervals, and there is missing data between the two adjacent time intervals. Utilization data is the only data that one can get on computer systems.

Using the EM-algorithm (a popular method for iterative calculation of the maximum likelihood estimate (MLE) based on the observed data) that is presented in [8], [9], it is demonstrated in [10] that the numerical MLEs of the intensity of the incoming flow of requests can be found based on the utilization data. However, the algorithm itself and its implementation are rather complicated. Still, under the assumptions of [10], it is possible to develop a simple probabilistic model of the system utilization factor (UF) that can serve as a basic tool for synthesizing and then analyzing the intensity estimation algorithm.

The purpose of this study is to obtain a MLE of the intensity of the incoming flow of requests, which allows a much simpler practical implementation compared to common analogues, and then to find its probabilistic and statistical characteristics. They can be evaluated through their comparison with the synthesized estimation algorithm simulation results.

2 Incoming Flow Intensity Estimate

The presented model of the probability density of the random UF variable is made of a mixture of discrete and uniform continuous probability densities. It has been studied in detail and substantiated in a number of studies taking the form [11]:

$$w(u) = P_0 \delta(u) + P_{01} + P_1 \delta(u - 1), \quad (1)$$

$$\begin{aligned}
P_0 &= P(U = 0), \quad P_{01} = P(0 < U < 1), \quad P_1 = P(U = 1), \\
P_0(\lambda, \mu) &= [\mu \exp(-\lambda(1-d)) - \lambda \exp(-\mu(1-d)) + \lambda - \mu] / (\lambda - \mu)(1 - \exp(-\lambda)), \\
P_1(\lambda, \mu) &= \lambda(\exp(-\mu) - \exp(-\lambda)) / (\lambda - \mu)(1 - \exp(-\lambda)), \\
P_{01}(\lambda, \mu) &= \frac{\exp[-\mu(1-d)] - \exp[-\lambda(1-d)] - \lambda \exp(-\mu) + \mu \exp(-\lambda)}{(\lambda - \mu)[1 - \exp(-\lambda)]}.
\end{aligned}$$

Only the weight parameters of this mixture depend on the intensity of the incoming flow λ , the intensity of service μ , and the fraction of the time of the UF calculating d . It can be shown that they contain complete information about the value of λ and, in the considered range of its values under the fixed μ and d , they can be approximated with high accuracy by second-order polynomials.

It is possible to estimate λ from the observed data

- using analytical expressions for the probabilities P_0 , P_{01} , P_1 finding the estimates of the parameter λ for each of them, and then – their weighted sum;
- by the method of moments;
- by the maximum likelihood method.

The first two listed methods are actually the implementation of the method of moments. Therefore, they have all the disadvantages inherent in this method [12]. When they are applied, the estimates can be found easily, but they are extremely sensitive to the fluctuations in the observed data and, accordingly, to the accuracy of estimating the probabilities P_0 , P_{01} , P_1 . In view of this, in our work, the MLE will be used as an estimate of the intensity λ , taking into account its many advantages. The MLE is found as the position of the absolute maximum of the likelihood function $L(\lambda)$. This estimate is sufficient, and also asymptotically (with an increasing sample size) efficient, unbiased, and normal [13], [14].

In [11], it was shown that the observed data (the number of the zero, intermediate and unit values of utilization – N_0 , N_{01} and N_1 , respectively) have a polynomial distribution. As a result, the likelihood function $L(\lambda)$ and its logarithm $M(\lambda) = \ln L(\lambda)$ for the data set with the volume $N = N_0 + N_{01} + N_1$ allows the following representation:

$$L(\lambda) = P_0^{N_0}(\lambda) P_{01}^{N_{01}}(\lambda) P_1^{N_1}(\lambda), \quad (2)$$

$$M(\lambda) = N_0 \ln P_0(\lambda) + N_{01} \ln P_{01}(\lambda) + N_1 \ln P_1(\lambda), \quad (3)$$

Taking into account, following [10], the possible low values of the parameter λ , to find the MLE one can apply the quadratic approximation of the logarithm of the likelihood function.

In Fig. 1, as an illustration, there are presented the diagrams of the logarithm of the likelihood function $M(\lambda)$ for the one of the realizations obtained by simulating the system with the real value of the intensity of the incoming flow of requests

$\lambda_0 = 1$. The dependences $M(\lambda)$ are plotted according to the formula (3) using the obtained exact relations for P_0 , P_{01} , P_1 (the curve 1) and the quadratic approximation of the likelihood function (curve 2).

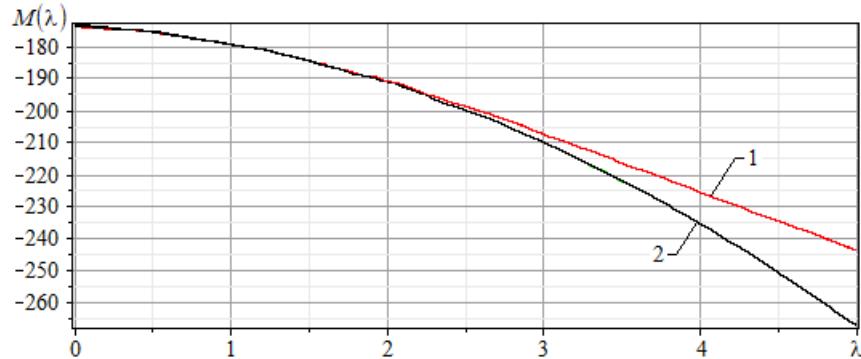


Fig. 1. The form of the logarithm of the likelihood function (3) for $\lambda_0 = 1$ and one of the simulation realizations plotted following the exact formulas for P_0 , P_{01} , P_1 (1) (the curve 1) and its quadratic approximation applied (the curve 2)

Fig. 1 demonstrates that these functions change smoothly in the considered range of values of the parameter λ . This allows concluding that simple approximation formulas for estimating λ can be reasonably accurate. In addition, it can be noted that for this particular realization of the observed data, the MLE value λ (the position of the $M(\lambda)$ maximum) differs from the real value of the parameter λ_0 .

The explicit analytical expression for the MLE, found from the solution of the probability equation for the chosen parameter values ($\mu = 10$, $d = 0.05$), takes the form:

$$\lambda_m \approx (22.37N_0 - 1.25N_1 - 18.94N)/(-2.42N_0 + 0.0091N_1 + 3.34N). \quad (4)$$

This relation is given for the fixed parameters μ and d due to the cumbersomeness of the general formula. The value obtained by replacing the frequencies N_0/N and N_1/N in (3) by the probabilities $P_0(\lambda, \mu)$, $P_1(\lambda, \mu)$ (1), on average, does not differ significantly from the real value of λ due to the good accuracy of the quadratic approximation of probabilities.

Statistical characteristics of the MLE λ_m can be determined by averaging according to the multinomial distribution (3) that the sample of the observed data (UF values) obeys:

$$P(N_0 = n_0, N_{01} = n_{01}, N_1 = n_1) = \frac{n!}{n_0! n_{01}! n_1!} P_0^{n_0}(\lambda, \mu) P_{01}^{n_{01}}(\lambda, \mu) P_1^{n_1}(\lambda, \mu).$$

Table 1 shows the sample means $\bar{\lambda}_m$ and mean-square deviations σ of the MLE (3) resulting from the AnyLogic software-based [15] simulation including 1000 runs of 200 one-second observation intervals at $\mu = 10$, $d = 0.05$. The parameters are specifically chosen for the comparison with the results from [10]. The values in the Table 1 are given with errors calculated in the $\pm 2\sigma$.

Table 1. The real values of the parameter λ , sample means $\bar{\lambda}_m$ and sample mean-square deviations σ of the MLE λ_m .

λ	0.2	0.5	1.0	1.5	2.0	2.2
$\bar{\lambda}_m$	0.190 ± 0.006	0.520 ± 0.009	1.01 ± 0.01	1.48 ± 0.014	1.99 ± 0.018	2.215 ± 0.0019
σ	0.097 ± 0.004	0.140 ± 0.006	0.160 ± 0.008	0.21 ± 0.06	0.27 ± 0.017	0.30 ± 0.02

In Table 1, one can see that the estimate has a small bias and a relatively small mean square error. The form of its probability density changes with λ . Since all the expansions in the approximations used to obtain (3) have been performed in the neighborhood of $\lambda = 1$, then, at this point, the characteristics of the approximate MLE should coincide with the properties of a reliable estimate. Indeed, according to the simulation results, the MLE distribution coincides with the Gaussian one with a significance level of 0.57. With a change in λ , the significance value decreases substantially.

In Fig. 2, there are

- the relative error $\delta_1(\lambda) = (\bar{\lambda}_m - \lambda)/\lambda$ expressed represented as rectangles,
- the relative mean-square error $\delta_2(\lambda) = 0.1\sigma/\lambda$ represented as circles,

both are expressed as percentage and dependent on λ .

The given estimate characteristics can be compared with the results [10] obtained using the Markov approach [16] to the synthesis of the likelihood function and the EM algorithm for determining the incoming flow intensity MLE at intervals. It follows from this comparative analysis that the proposed approach, being much simpler, has similar (at least not worse) accuracy characteristics. In addition, it is possible to write down an analytical expression for the desired MLE, which also simplifies its technical implementation.

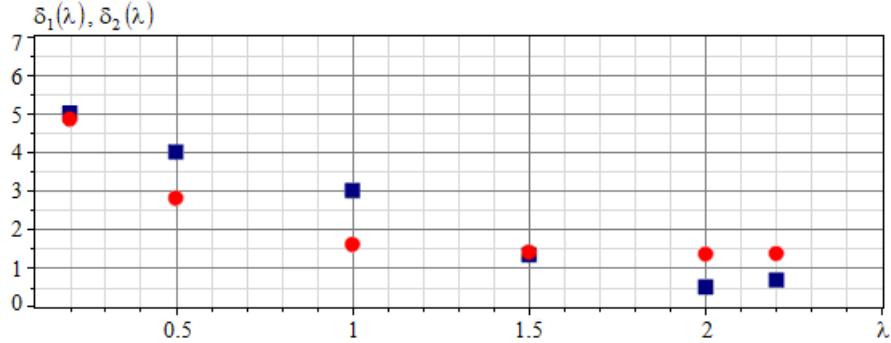


Fig. 2. Expressed as percentage, the relative error (rectangles) and the relative mean-square error (circles) of the incoming flow intensity MLE, in their dependence on λ

3 Conclusion

The presented approximate analytical probabilistic model (1) for the utilization factor of the M|M|1 queuing system in the form of a distribution mixture 1) is quite simple; 2) has high accuracy; 3) is confirmed by the simulation results. It can serve as a basic tool for synthesizing the system parameter estimation algorithms based on the available data on the application of this model.

The simple maximum likelihood estimate synthesized with its help is unbiased and relatively easy to implement in practice. The possibility of applying this estimate for determining the intensity of the incoming flow of requests by the system utilization factor at short observation time intervals is confirmed by the results of simulation.

Thus, the approach realized includes the following stages: 1) simulation of the initial problem; 2) developing the theoretical model of the observed data following the results of the simulation; 3) synthesizing approximate MLE according to the simulation model; 4) analyzing the statistical characteristics of the estimate in terms of its simulation model. This approach allows producing relations for the estimate of the intensity of the incoming flow of requests that are quite accurate and relatively simple, compared with the ones introduced in the common studies.

The approach and results presented in this study are partially valid for a non-uniform Poisson flow of requests arriving for servicing. Since only a small part of the observation interval is used to measure the UF, the flow in this interval can often be considered uniform.

In previous studies, the conditions for the validity of the model considered in the work were formulated. When they are realized, no more than one event is observed associated with the beginning or end of servicing the request in a short sub-interval of each one-second interval. If the formulated conditions are satisfied, then the model remains valid for several demand service devices.

The possibility of interfacing the simulation in the AnyLogic environment with the reinforcement learning (the Bonsai platform) opens up the prospects for obtaining stable functional structures in the problem under consideration in the case when a priori assumptions about the initial parameters are violated.

Acknowledgements

This research was financially supported by the Russian Science Foundation (research project No. 20-61-47043).

References

1. Thiruvaiyaru, D., Basawa, I.V., Bhat U.N.: Estimation for a Class of Simple Queueing Networks. *Queueing Systems* 9, 301–312 (1991)
2. Basawa, I.V., Bhat, U.N., Lund, R.: Maximum Likelihood Estimation for Single Server Queues from Waiting Time Data. *Queueing Systems* 24, 155–167 (1996)
3. Ross, J.V., Taimre, T., Pollett, P.K.: Estimation for Queues from Queue Length Data. *Queueing Systems* 55, 131–138 (2007)
4. Pant, A.P., Ghimire, R.P.: M(t)/M/1 Queueing System with Sinusoidal Arrival Rate *J. Inst. Eng.* 11, 120–127 (2015)
5. Thiruvaiyaru, D., Basawa, I.V.: Empirical Bayes Estimation for Queueing Systems and Networks. *Queueing Systems* 11, 179–202 (1992)
6. Choudhury, A., Basak, A.: Statistical Inference on Traffic Intensity in an M/M/1 Queueing System. *Int. J. Management Science and Engineering Management* 13, 274–279 (2018)
7. Dharmaraja, S., Trivedi, K., Logothetis, D.: Performance Modeling of Wireless Networks with Generally Distributed Handoff Interarrival Times. *Comp. Comm.* 26, 1747–1755 (2003)
8. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. Springer, New York (2001)
9. McLachlan, G.J., Krishnan, T.: *The EM Algorithm and Extensions*. Wiley-Interscience, Cambridge (2008)
10. Li, C., Okamura, H., Dohi, T.: Parameter Estimation of M-t/M/1/K Queueing Systems with Utilization Data. *IEEE Access* 7, 42664–42671 (2019)
11. Zyulkov, A., Kutoyants, Y., Perelevskiy, S., Korableva, L.: Single Channel Queueing System Utilization Factor Model. *J. Phys. Conf. Ser.* 2388, 1–7 (2022)
12. Cramer, G.: *Mathematical Methods of Statistics*. Princeton University Press, Princeton (1951)
13. Ibragimov, I.A., Has'minskii, R.Z.: *Statistical Estimation. Asymptotic Theory*. Springer, New York (1981)
14. Van Trees, H.L., Bell, K.L., Tian, Z.: *Detection, Estimation and Modulation Theory, Part I. Detection, Estimation, and Filtering Theory*. Wiley, New York (2013)
15. Borshchev, A., Grigoryev, I.: *The Big Book of Simulation Modeling. Multimethod Modeling with AnyLogic 8*. AnyLogic North America, Oakbrook Terrace (2020)
16. Okamura, H., Dohi, T., Trivedi, K.S.: Markovian Arrival Process Parameter Estimation with Group Data. *IEEE/ACM Trans. on Networking* 17, 1326–1339 (2009)

Statistical Characteristics of the Adjacent Information Signals Amplitude Ratio

*Oleg Chernoyarov¹, Alexey Glushkov², Vladimir Litvinenko³, Alexandra Salnikova⁴, and Kaung Myat San⁵

¹National Research University “MPEI”, Moscow, Russia
^{1*}chernoyarovov@mpei.ru

²Zhukovsky-Gagarin Air Force Academy, Voronezh, Russia
²al.nk.glushkov@gmail.com

³Voronezh State Technical University, Voronezh, Russia
³vl.pt.litvinenko@gmail.com

⁴National Research University “MPEI”, Moscow, Russia
⁴al.vl.salnikova@mail.ru

⁵National Research University “MPEI”, Moscow, Russia
⁵kmyatsan@mail.ru

* Corresponding Author

Abstract. The ratio of the amplitudes of adjacent information symbols at the output of the demodulator is used in the processing of the differential amplitude and phase shift keyed signals (DAPSK), when the phase shift keyed signals (PSK and DPSK) are detected, as well as to determine the quality of a working communication channel, to measure the channel signal-to-noise ratio and in a number of other tasks. It is proposed to use the logarithm of the adjacent information symbol amplitude ratio as a decision determining statistics because its probabilistic characteristics does not depend on the level of interferences in the absence of a signal, while in the presence of a signal, its properties depend only on the signal-to-noise ratio. The probabilistic characteristics of the proposed statistics and the possibility of their practical use are considered. The necessary calculation relations are obtained; statistical simulation is carried out for PSK signal.

Keywords: Phase-shift keyed signal, decision determining statistics, adjacent symbol amplitude ratio, probability of exceeding the threshold, signal detection, false alarm and missing probabilities.

1 Introduction

Phase-shift (PSK) and frequency-shift keyed (FSK) signals are characterized by a constant symbol amplitude, while at the output of a narrow-band transmission path, the pulse shape differs from a rectangular one, but in this case, the receiver response level is proportional to the amplitude of the received symbol. For the amplitude and phase-shift keyed (APS) signals, the symbol amplitudes change, but their ratio a

remains unchanged (for example, 3 or 1/3 [1], [2] or other options 2.5, 1.4 [3-5]) at different gains of the receiving path. In these applications, the ratio of the adjacent symbol amplitudes can be used to demodulate a signal or control a demodulator.

When a signal is detected (for example, the PSK one) [6-8], it is necessary to establish its presence in a noisy channel. In the absence of a signal, the responses of the amplitude estimation channel obey a Rayleigh probability distribution [9] and the ratios of their neighboring values are characterized by a significant statistical variation. In the presence of a signal, this variation decreases with the increasing signal-to-noise ratio (SNR), which can be used to detect the presence of a signal.

The efficiency of detectors depends on the amount of a priori information about the levels and properties of signals and noise [9-11]. In this case, the actual task is to develop detection algorithms providing a false alarm probability that is independent of the noise level. It can be solved using the amplitude ratio of the adjacent symbols. Using a sampling of the amplitude ratios ensures a specified reliability of the obtained solutions and allows controlling the quality of the operating communication channel and estimating the SNR.

A similar approach using the ratio of the energy parameters of the two adjacent symbols can be used in the processing and demodulation of statistical signals with relative modulation formats [12].

The choice of the logarithm of the ratio of the amplitudes of the adjacent symbols as a sign of the presence of a signal or a characteristic of the levels of relative amplitude keying is due to the simplicity of implementation and the absence of the need to interfere with the data flow.

Thus, the study of the probabilistic properties of the ratio of the symbol amplitudes is quite relevant.

2 Signal and Noise Models

Discrete signals with an amplitude that does not depend on the transmitted information are considered, these are PSK and FSK signals taking the form

$$s(t) = S \cos[2\pi f_0 t + \psi(t)]$$

with the constant amplitude S , the carrier frequency f_0 , and current phase $\psi(t)$. At the output of the transmission path, the pulse shape is distorted, but the magnitude of the response at the end of the symbol in the absence of interference remains independent of the transmitted messages [6]. The appearance of interference distorts the amplitudes of the adjacent symbols, and their dissimilarity, estimated by the ratio of these amplitudes, can be used to determine the SNR in the communication channel without interfering with the data transmission process.

When there is Gaussian noise $n(t)$ or there are both noise and signal at the demodulator input, the response samples ξ_k of the amplitude estimation channel (where k is the number of the current symbol) at the end of each symbol reception

is determined by the clock synchronization signal. In the presence of noise only, such samples obey the Rayleigh law of distribution with the probability density [7-11]

$$w_n(x) = x \exp(-x^2/2\sigma_n^2)/\sigma_n^2, \quad x \geq 0. \quad (1)$$

But they obey the Rice law of distribution with the probability density [7-11]

$$w_s(x) = x \exp[-(x^2 + S^2)/2\sigma_n^2] I_0(xS/\sigma_n^2)/\sigma_n^2, \quad x \geq 0 \quad (2)$$

in the presence of both information signal and noise. Here σ_n^2 is the noise dispersion at the receiver input and $I_0(x)$ is the zero-order modified Bessel function.

3 Symbol Amplitude Ratio

The ratio of the amplitudes of the adjacent symbols at the output of the amplitude estimation channel is

$$Y_k = \xi_k / \xi_{k-1}. \quad (3)$$

But, as the decision determining statistics it is more expedient to use the logarithm of this magnitude

$$G_k = \log_a Y_k = \log_a (\xi_k / \xi_{k-1}). \quad (4)$$

For the DAPSK signals, the base a of the logarithm can be chosen equal to the ratio of the maximum and minimum symbol amplitudes (for example, $a=3$ [1], [2]). As for the signals with a constant symbol amplitude, it is advisable to take the logarithm in (4) as the natural one, that is,

$$G_k = \ln(\xi_k / \xi_{k-1}). \quad (5)$$

Decisions about the received signals (for example, about their detection) can be made by the sampling G_k (5), of the size N , and $k = \overline{1, N}$. It is obvious that the neighboring values Y_k and Y_{k-1} are dependent, since they include the same value ξ_{k-1} . To ensure the independence of the amplitude ratios, they must be determined for the individual pairs of ξ_k samples, but in this case, the sampling size of G_k samples is halved. For independent values of G_k , a binomial probability

distribution can be used, and for dependent ones a model based on a simple Markov chain [13] is true.

4 The Properties of the Amplitude Ratio in the Absence of a Signal

According to [14], the probability density $u(y)$ of the ratio (3) of the random variables ξ_k and ξ_{k-1} with the same probability densities $w(x)$ (1) or (2) is

$$u(y) = \int_0^\infty w(x)w(yx)|x|dx \quad (6)$$

and then, correspondingly, the probability density $f(g)$ of the values G_k (5) can be obtained as

$$f(g) = \exp(g)u(\exp(g)). \quad (7)$$

Accounting for (1) and based on (6), and in the presence of noise only, after calculating the integral, the probability density $u_n(y)$ of the ratio (3) takes the form

$$u_n(y) = 2y/(1+y^2)^2, \quad y \geq 0 \quad (8)$$

and from (7) and (8), for $f_n(g)$, one gets respectively:

$$f_n(g) = 2\exp(2g)/[1+\exp(2g)]^2.$$

One can see that these probability densities – $u_n(y)$ and $f_n(g)$ – do not depend on the noise dispersion, their diagrams are presented in Fig. 1a and Fig. 1b.

The mean value of the ratio Y_k (3) is $Y_{\text{mean}} = \int_0^\infty yu_n(y)dy = \pi/2$, while the variance tends to infinity. For the logarithmic measure G_k (5), the mean value is equal to 0, while the variance is

$$\sigma_G^2 = \int_{-\infty}^\infty g^2 f_n(g)dg = \pi^2/12 = 0.822. \quad (9)$$

The curve $f_n(g)$ is symmetric to the y -axis and more feasible for use in

comparison with $u_n(y)$. Positive values of g correspond to the region of relations $y > 1$, and the negative values of g correspond to $y < 1$, respectively.

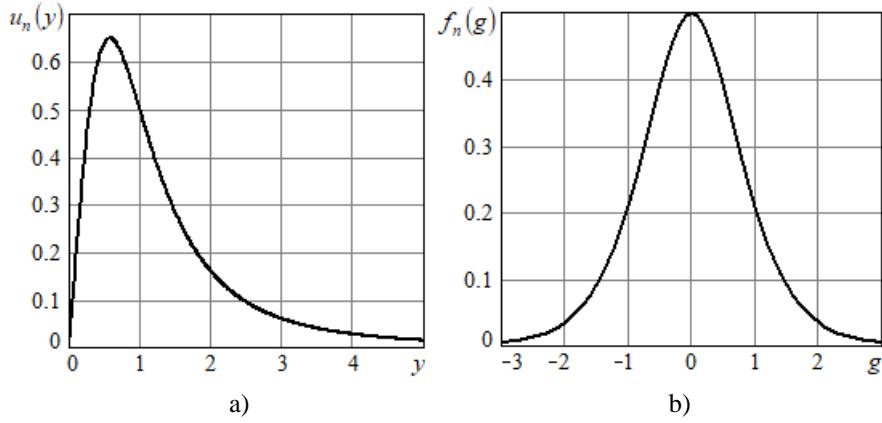


Fig. 1. The probability densities of the adjacent symbol amplitude ratio (a) and its logarithm (b) in the absence of a signal

Now it is time to determine the probability $p_n(b)$ that the modulus of the logarithm of the ratio of amplitudes $|G_k|$ exceeds the specified threshold b :

$$p_n(b) = 1 - \int_{-b}^b f_n(g) dg = 2 / [1 + \exp(2b)] . \quad (10)$$

This dependence is demonstrated in Fig. 2. As it can be seen, at $b < 0.5$, the probability (10) is quite high, meaning that it can serve as an indicator of the absence of a signal that does not depend on the noise dispersion.

Since the neighboring samples of the amplitude ratio Y_k and Y_{k-1} are dependent, it is necessary to consider their joint probability density.

Now, the notation is

$$Z_k = Y_{k-1} Y_k , \quad (11)$$

then, accounting for (3),

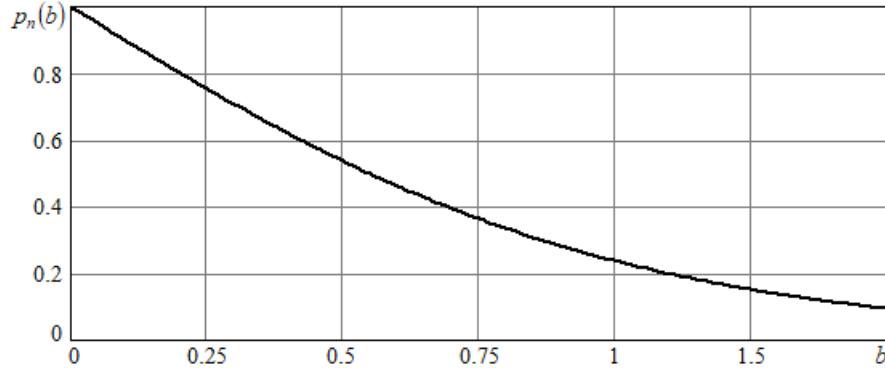


Fig. 2. The probability of exceeding the threshold by the modulus of the logarithm of the adjacent symbol ratio in the absence of a signal

$$Z_k = \frac{\xi_{k-1}}{\xi_{k-2}} \frac{\xi_k}{\xi_{k-1}} = \frac{\xi_k}{\xi_{k-2}} . \quad (12)$$

The magnitude Z_k is, on the one hand, the product of the amplitude ratio, and, on the other hand, it is the amplitude ratio with the probability density (8). According to [14], the probability density of the product $Z = Y_1 Y_2$ of the random variables Y_1 and Y_2 with the joint probability density $w_n(y_1, y_2)$ is

$$w(z) = \int_{-\infty}^{\infty} w\left(y_1, \frac{z}{y_1}\right) \frac{1}{|y_1|} dy_1 . \quad (13)$$

For the conditional probability density $w_n(y_k | y_{k-1})$, based on (8) and accounting for (11) and (13), one can write

$$w_n(y_k | y_{k-1}) = 2y_{k-1}^2 y_k / \left[1 + (y_{k-1} y_k)^2 \right]^2 . \quad (14)$$

Then for the joint probability density of the amplitude ratio in the absence of a signal, accounting for (8), (14), one gets:

$$w_n(y_{k-1}, y_k) = u_n(y_{k-1}) w_n(y_k | y_{k-1}) = \frac{4y_{k-1}^3 y_k}{(1 + y_{k-1}^2)^2 [1 + (y_{k-1} y_k)^2]^2} . \quad (15)$$

It easy to see that

$$\int_0^\infty \int_0^\infty w_n(y_{k-1}, y_k) = 1$$

and also, following from (13), (14), the probability density of the random variable (11), (12) takes the form

$$w_n(z) = \int_0^\infty w_n\left(y_{k-1}, \frac{z}{y_{k-1}}\right) \frac{1}{|y_{k-1}|} dy_{k-1} = \frac{2z}{(1+z)^2}$$

and coincides with that in (8).

According to (7), (15) the joint probability density of the neighbouring logarithmic measures G_{k-1} , G_k (5) is

$$w_n(g_{k-1}, g_k) = \frac{4 \exp(4g_{k-1}) \exp(2g_k)}{[1 + \exp(2g_{k-1}) \exp(2g_k)]^2 [1 + \exp(2g_{k-1})]^2}, \quad (16)$$

its three-dimensional diagram is shown in Fig. 3.

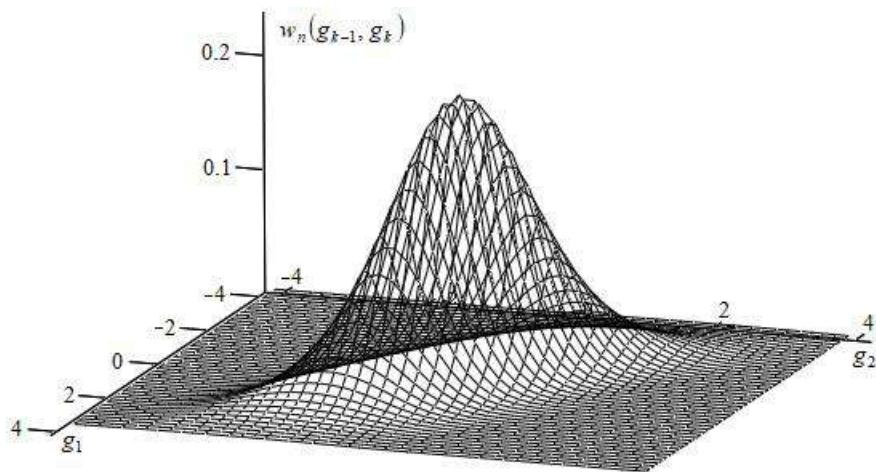


Fig. 3. Two-dimensional probability density of the logarithm of the ratio of amplitudes (5) in the absence of a signal

It follows from (16) that the correlation coefficient of the values G_{k-1} and G_k (5) is equal to -1 , and the sequence of G_k forms a simple Markov chain.

5 The Properties of the Amplitude Ratio in the Presence of a Signal

In the presence of a signal, the probability density $w_s(x)$ of the signal amplitudes is described by the Rice formula (2) and the probability density $u_s(y)$ of their ratio (3), following (6) is

$$u_s(y) = \int_0^\infty w_s(x)w_s(yx)|x|dx. \quad (17)$$

After setting the SNR as

$$h^2 = S^2/2\sigma_n^2. \quad (18)$$

and changing the variables $v = x/\sigma_n$, from (17), one gets

$$u_s(y) = \int_{-\infty}^\infty yv^3 \exp\left[-(v^2 + (yv)^2 + 4h^2)/2\right] I_0(vh\sqrt{2}) I_0(yvh\sqrt{2}) dv. \quad (19)$$

Based on (7), the expression for the probability density of the logarithm of the adjacent symbol amplitude ratio (5) is now produced:

$$f_s(g) = \exp(g) u_s(\exp(g)). \quad (20)$$

The integrals (19) and (20) are calculated numerically only. But, if $h \gg 1$ (18), then using Gaussian approximation of the Rice distribution (2) and following (19), (20), one obtains

$$u_s(y) = h(1+y) \exp\left[-h^2(1-y)^2/(1+y^2)\right] / \sqrt{\pi(1+y^2)^3}, \quad (21)$$

$$f_s(g) = \frac{h \exp(g)(1+\exp(g))}{\sqrt{\pi(1+\exp(2g))^3}} \exp\left[-h^2(1-\exp(g))^2/(1+\exp(2g))\right]. \quad (22)$$

As it can be seen, the one-dimensional probability densities $u_s(y)$, $f_s(g)$ depend only on SNR (18). Their diagrams are shown in Fig. 4a and fig. 4b, respectively, for the case when $h^2 = 10$. The solid lines describe the exact dependences (17), (20), and the dashed lines – the approximate ones (21), (22), and that, as the graphs demonstrate, have satisfactory accuracy for $h^2 \geq 10$.

According to the results of numerical calculations, the mean value of the random

variable (5) can practically be taken equal to zero, then its variance is determined by the expression

$$\sigma_G^2 = \int_{-\infty}^{\infty} g^2 f_s(g) dg . \quad (23)$$

The dependence of σ_G^2 (23) on SNR h^2 (expressed in dB) is shown by the solid line in Fig. 5, with the dashed line demonstrating the value of the variance σ_G^2 (9) in the absence of a signal being 0.822. One can see that an increase in the signal level leads to a decrease in the variance of the logarithm of the adjacent symbol amplitude ratio, and that can be used to detect a signal or to estimate SNR in an operating channel.

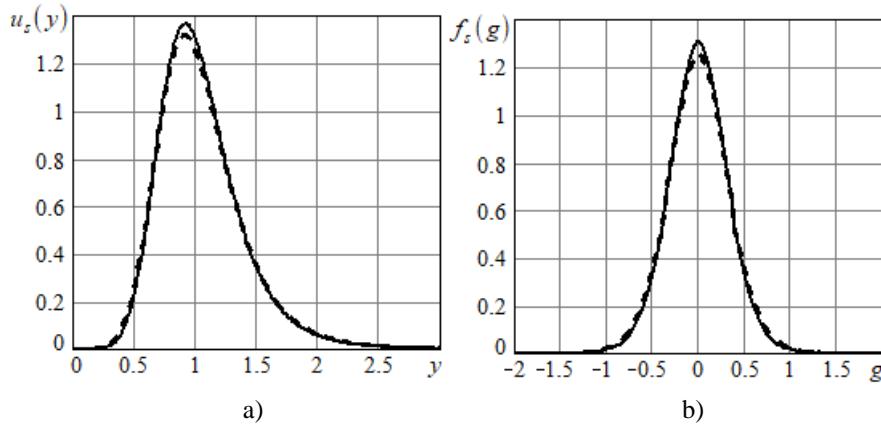


Fig. 4. The probability densities of the adjacent symbol amplitude ratio (a) and its logarithm (b) in the presence of a signal

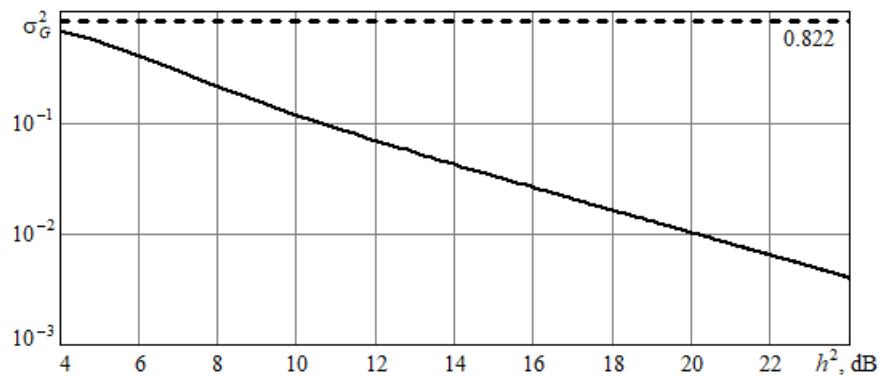


Fig. 5. The variance of the logarithm of the adjacent symbol ratio

It is of interest to calculate the probability $p_s(b)$ that the modulus of the logarithm of the ratio of the amplitudes $|G_k|$ will exceed the specified threshold b in the presence of a signal. Taking into account (22), one gets

$$p_s(b) = 1 - \int_{-b}^b f_s(g) dg . \quad (24)$$

The dependences (24) for various SNR (18) are plotted in Fig. 6. There, the curve 1 corresponds to $h^2 = 10$, the curve 2 – to $h^2 = 20$, the curve 3 – to $h^2 = 30$, the curve 4 – to $h^2 = 40$. The dashed line represents the curve in the absence of a signal (Fig. 2).

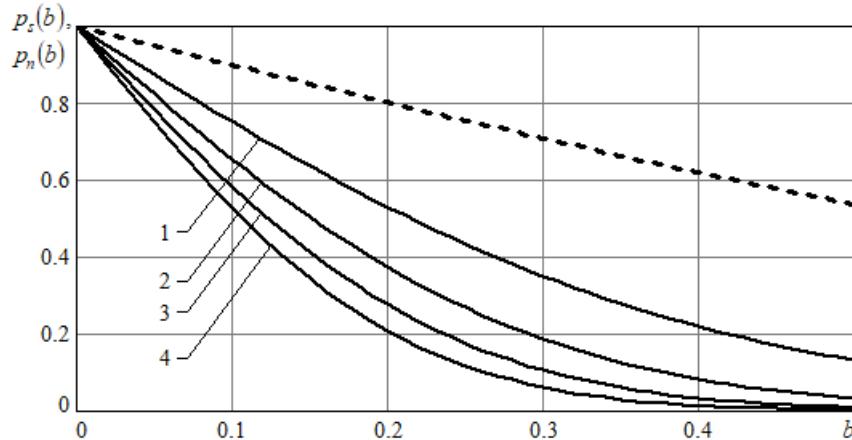


Fig. 6. The probability of exceeding the threshold by the modulus of the logarithm of the adjacent symbol ratio in the presence and absence of a signal

It follows from (19), (20), (24) and Fig. 6 that the probability of exceeding the threshold decreases with the increase in the SNR, and it does not depend on the noise level, meaning that its estimate can be used to control the quality of the communication channel.

In the presence of a signal, the neighboring values of the amplitude ratios are interdependent, and introducing their product

$$Z_k = Y_{k-1}Y_k , \quad (25)$$

by analogy with (13)-(15), one can write the expression for the conditional

probability density

$$w_s(y_k|y_{k-1}) = \frac{hy_{k-1}(1+y_{k-1}y_k)}{\sqrt{\pi[1+(y_{k-1}y_k)^2]^3}} \exp\left[-\frac{h^2(1-y_{k-1}y_k)^2}{1+(y_{k-1}y_k)^2}\right].$$

while for the joint probability density of the amplitude relations one gets

$$w_s(y_{k-1}, y_k) = \frac{h^2 y_{k-1}(1+y_{k-1})(1+y_{k-1}y_k)}{\pi \sqrt{[1+y_{k-1}^2]^3 [1+(y_{k-1}y_k)^2]^3}} \exp\left[-h^2\left(\frac{(1-y_{k-1}y_k)^2}{1+(y_{k-1}y_k)^2} + \frac{(1-y_{k-1})^2}{1+y_{k-1}^2}\right)\right]. \quad (26)$$

It is easy to see (for example, by the numerical integration) that $\int_0^\infty \int_0^\infty w_s(y_{k-1}, y_k) dy_{k-1} dy_k = 1$ and the probability density of the random variable (25) which can be found as

$$u_s(z) = \int_0^\infty w_s\left(y_{k-1}, \frac{z}{y_{k-1}}\right) \frac{1}{|y_{k-1}|} dy_{k-1}$$

coincides with (21).

From (26), it follows that the joint probability density of the logarithmic measures G_{k-1} and G_k , accounting for (20), is

$$\begin{aligned} w_s(g_{k-1}, g_k) &= \frac{h^2}{\pi} \frac{\exp(2g_{k-1} + g_k)[1+\exp(g_{k-1})][1+\exp(g_{k-1} + g_k)]}{\sqrt{[1+\exp(2g_{k-1})]^3 [1+\exp(2(g_{k-1} + g_k))]^3}} \times \\ &\times \exp\left[-h^2\left(\frac{(1-\exp(g_{k-1} + g_k))^2}{1+\exp(2(g_{k-1} + g_k))} + \frac{(1-\exp(g_{k-1}))^2}{1+\exp(2g_{k-1})}\right)\right]. \end{aligned} \quad (27)$$

The three-dimensional diagram of $w_s(g_{k-1}, g_k)$ (27) at $h^2 = 40$ (that is, 16 dB) is shown in Fig. 7.

It is demonstrated there that the presence of a signal significantly “narrows” the surface compared to that shown in Fig. 3 (that is, the variance of the logarithmic ratio (5) decreases, as one can see in Fig. 5). But, it should be noted that the neighboring values of G_{k-1} and G_k remain correlated. For example, at $h^2 = 40$, their correlation coefficient $R_G = \langle G_{k-1}G_k \rangle$ is equal to -0.807 . However, the value of R_G decreases with the increase in the SNR and tends to zero when $h \rightarrow \infty$.

Thus, in the presence and absence of a signal, adjacent amplitude ratios G_{k-1} and G_k are statistically interrelated, while the correlation coefficient in the area of

working SNRs is in the range from -0.807 to -0.5 .

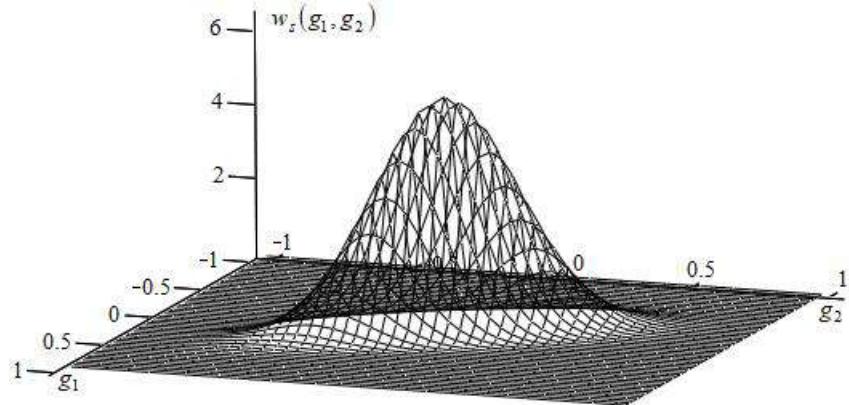


Fig. 7. Two-dimensional probability density of the logarithm of the ratio of the amplitudes (5) in the presence of a signal

6 The Sequence of Values of the Logarithm of the Adjacent Symbol Amplitude Ratio

In the simplest case, the decision about the properties of the signal should be taken even by one value (sample) of the measure (5). For example, when a signal is detected, the following algorithm can be used: the signal is absent if $G_k > b$, and if $G_k \leq b$, then the signal is detected. In this case, following (10), the probability of the false alarm P_F is

$$P_F = 1 - p_n(b). \quad (28)$$

and, based on (24), the signal missing probability P_M is, correspondingly

$$P_M = p_s(b). \quad (29)$$

Thus, for the specified value of P_F , the required threshold b is determined from the equation (28), and then, from (29), the value of P_M is determined for the selected SNR h^2 (32). For example, when $P_F = 0.1$, then, following (28) and taking into account (10) and Fig. 2, one gets $b \approx 0.1$, while from (24), at $h^2 = 272$ (24.3 dB), according to (29), one gets $P_M \approx 0.1$.

It means that it is impossible to carry out a reliable detection based one sample, and thus it is necessary to use the sampling G_k , $k = \overline{1, N}$ for the sequence of $N + 1$

information symbols (where the value of N should be taken as the maximum possible).

To estimate the signal properties one can use the sample estimate of the variance (23):

$$\sigma_G^2 = \frac{1}{N-1} \sum_{k=1}^N G_k^2,$$

based on which as well as on Fig. 5, the decisions can be made. High estimation accuracy is ensured by a large size of the sample $N \gg 1$. It should be noted that the estimates of the correlation coefficient are unsuitable for this, since its significant module decrease occurs at a high SNR.

If the values of G_k are independent (for example, only even samples are selected, while the sample size is halved), then the probability $P_n(M_0)$ that in a sampling made of N samples, the threshold b is exceeded no more than M_0 times, is equal to in the absence of a signal [15]

$$P_n(M_0) = \sum_{m=0}^{M_0} \binom{N}{m} p_n^m (1-p_n)^{N-m}. \quad (30)$$

where $p_n = p_n(b)$ following (10). This probability decreases with M_0 which allows providing the required level of false alarms.

A similar probability $P_s(M_0)$, in the presence of a signal, can be determined by replacing p_n in (30) by $p_s = p_s(b)$ from (24) and then the signal missing probability can be found.

In the full sampling G_k , the neighboring values are strongly correlated and the binomial model (30) becomes approximate. In this case, it is advisable to describe the sampling using a simple Markov chain. Two events are considered: the sample that is lower than the specified threshold b (index 0) and higher than it (index 1), with the four combinations of these events including the corresponding transition probabilities p_{ij} , $i, j = 0, 1$ from the event i to event j for the previous, that is G_{k-1} , and the current, that is G_k , values of the samples. The matrix of transition probabilities has the form [13]

$$[p_{ij}] = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\beta & \beta \end{bmatrix}.$$

and contains two parameters α and β calculated based on the two-dimensional probability density $w_s(g_{k-1}, g_k)$ (27). The possibilities of applying the Markov model for the solution of the problem under consideration require additional

research.

Thus, the considered approaches enable the researcher to make decisions about the presence of a signal and the SNR value in an operating channel without the need to estimate the noise and signal levels.

7 Conclusion

It is proven that the logarithm of the adjacent information symbol amplitude ratio is an effective measure for estimating the properties of the received discrete signal and the communication channel as a whole, and it does not depend on the absolute level of the channel noise and is SNR determined only. For this purpose, the probabilistic properties of the proposed measure are considered, the necessary calculation relationships are obtained, and both one-dimensional and two-dimensional probability densities are found. It is demonstrated that the statistical characteristics of the introduced measure, as well as the probabilities of exceeding the specified threshold by the measure, regardless of the levels of noise and signal, can be obtained analytically. It is established that the variance of the logarithm of the amplitude ratio in the absence of a signal does not depend on the noise level and is equal to 0.822, while in the presence of a signal it decreases with the increase in the signal-to-noise ratio.

The properties of the sequence of the measure samples are considered, including taking into account the correlation of the neighboring values. Based on them, it is proposed to simulate the sequence of the samples of the logarithm of the amplitude ratio by a simple Markov chain.

The obtained probabilistic characteristics can be used when finding solutions for the problems of detecting a signal with a noise level-independent false alarm probability and in SNR estimating.

Acknowledgements

The work was supported by the Ministry of Education and Science of the Russian Federation (research project No. FSWF-2023-0012).

References

1. Sklar, B., Harris, F.J.: Digital Communications: Fundamentals and Applications. Pearson, London (2020)
2. Proakis, J.G., Salehi, M.: Digital Communications. McGraw-Hill Higher Education, New York (2007)
3. Liang, D., Ng, S.X., Hanzo, L. Near-Capacity Turbo Coded Soft-Decision Aided DAPSQSK/Star-QAM for Amplify-and-Forward Based Cooperative Communications. IEEE Trans. Comm. 61, 1080–1087 (2013)

4. Genç, F., Yengel, E., Savaşçıhabes, A., Ertuğ, Ö.: On the Optimum Ring Ratio Determination for 16-DAPSK Modulation in OFDM Systems. In: 2014 22nd Signal Processing and Communications Applications Conference, pp. 742–745. IEEE Press, Trabzon (2014)
5. Chernoyarov, O., Litvinenko, V., Kaung Myat San, Glushkov, A., Litvinenko, Y.: On the Demodulation of the ADPSK Signals. In: 14th International Conference “ELEKTRO2022”, pp. 1–5. IEEE Press, Krakow (2022)
6. Litvinenko, V.P., Litvinenko, Y.V., Matveev, B.V., Chernoyarov, O.V., Makarov, A.A.: The New Detection Technique of Phase-Shift Keyed Signals. In: 2016 International Conference on Applied Mathematics, Simulation and Modelling, pp. 355–358. Atlantis Press, Zhengzhou (2016)
7. Urkowitz, H.: Energy Detection of Unknown Deterministic Signals. Proceedings of the IEEE 55, 523–531 (1967)
8. Chernoyarov, O.V., Golpaiegani, L.A., Glushkov, A.N., Litvinenko, V.P., Matveev, B.V.: Digital Binary Phase-Shift Keyed Signal Detector. Int. J. Eng. Trans. A: Basics 32, 510–518 (2019)
9. Van Trees, H.L., Bell, K.L., Tian, Z.: Detection, Estimation and Modulation Theory, Part I. Detection, Estimation, and Filtering Theory. Wiley, New York: (2013)
10. Kay, S.: Fundamentals of Statistical Signal Processing: Detection Theory, Volume 2. Pearson, London (1998)
11. Levin, B.R.: Theoretical Principles of Radioengineering Statistics. Defense Technical Information Center, Virginia (1968)
12. Didkowsky, R.M., Pervuninskii, S.M., Bokla, N.I.: Basic Methods of Stochastic Signal Modulation. Doklady BGUIR 20, 50–55. (2013)
13. Kirkwood, J.R.: Markov Processes. CRC Press, Boca Raton (2015).
14. Grami, A.: Probability, Random Variables, Statistics, and Random Processes: Fundamentals & Applications. Wiley, New York (2019).
15. Johnson, N.L., Kotz, S., Balakrishnan, N.: Discrete Multivariate Distributions. Wiley-Interscience, New York (1997)

Predicting Severity Levels of Parkinson's Disease from Telemonitoring Voice Data

Aryan Vats^{1[0000-0003-0675-5095]}, Aryan Blouria^{2[0000-0003-3876-0884]} and Sasikala R.^{3[0000-0003-2012-3748]}

^{1, 2} Student at Vellore Institute of Technology, Vellore Tamil Nadu 632014, India

¹ avats101@gmail.com

² aryan.blouria2019@vitstudent.ac.in

³ School of Computer Science and Engineering, Vellore Institute of Technology, Vellore Tamil Nadu 632014, India

³ sasikala.ra@vit.ac.in

Abstract. Parkinson's disease is an age-related degenerative brain disorder that affects the nervous system causing symptoms such as tremors, stiffness, and slowing of movement that begin gradually and worsen over time. As severity increases, problems such as mental and behavioural changes, memory issues, depression and fatigue arise. As such, to be able to successfully diagnose and cure patients of Parkinson's, we need an accurate and reliable means to identify the severity of the disease. This paper proposes a method of severity prediction using neural networks, with the adequate number of severity levels backed by statistical proofs. This paper focuses on obtaining an accurate number of clusters using the Telemonitoring Voice Data set of patients, and through our experiments, we arrive at four severity levels. In the proposed approach, emphasis was given to consider more parameters for prediction and the exactness of the values obtained by our strategy to be better when contrasted with the precision acquired in past research. The system proposed can classify Parkinson patients into more severity levels than just 'severe' and 'non-severe' and consequently can help clinical professionals in medical care diagnose Parkinson Illness.

Keywords: Parkinson's disease, Severity Ranking, Unified Parkinson's Disease Rating Scale, Parkinson's Telemonitoring, Clustering, Neural networks.

1 Introduction

The Parkinson's disease is a progressive neurological condition occurring by the degeneration happening in the nerve cells of the brain that control movement by producing a chemical called dopamine. As these cells are impaired, they lose the ability to produce dopamine and hence the patient develops symptoms of Parkinson's disease. Although the exact causes are unknown, it is believed to be caused due to multiple factors such as genetics, environmental toxins etc. Symptoms of this disease include loss of movement and balance, tremors, depression etc. Parkinson Disease patients also show substantial difference in voice attributes, thus vocal recordings are an important noninvasive procedure for having diagnosis. Voice recordings passed through signal processing algorithms with regression methods forecast a value on the

Unified Parkinson's Disease Rating Scale (UPDRS) [1]. The analogy obtained from the work conducted in [2] suggests classifying the severity of PD into three levels based on the outputs obtained from the UPDRS score, giving to the concept of deriving severity levels for better diagnosis of PD. These levels set a baseline threshold for all voice features, defining a boundary. The use of Neural Networks for efficacious analysis of speech signals is not archaic, having techniques involving multiple layers stacked together to produce a classifier. This paper builds a classifier on the dataset worked in [1] and uses clustering algorithms to devise the number of levels. These clusters are visualized using t-SNE [3], giving an effective representation for the clusters generated using the highly dimensional data. This technique emphasizes on using total UPDRS and motor UPDRS along with the voice features, giving a higher dimensional set for the levels to classify rather than considering only scale values as done in [4].

2 Related Work

To come up with a reliable and accurate approach for predicting severity of Parkinsons' in a patient, a lot of research was undertaken. In [2], a goal of trying to determine severity levels based on the UPDRS scores was achieved with a classification into 'mild', 'moderate' or 'severe'. They made use of concepts involved in data analytics for their study, involving central tendency, dispersion measures and concordance. In [4] a neural net was implemented to predict the disease severity into two classes ('severe' and 'non-severe') setting a foundation for our methodology. A lot of weightage was given to UPDRS scores in prediction of the severity. Another approach for severity estimation is attempted in [5], the findings of which proved that acoustic analysis of speech signals had great potential in diagnosing Parkinson's. It used regression analysis made on regularized Random Forests to find statistical scores based on different speech tasks like words, sentences, texts, monologues and combination of them.

In [6], various machine learning approaches were employed to predict the UPDRS scores based on 16 voice features and the performance of the neural network was found to be desirable, when compared with Support Vector Regression, Decision Tree Regression and Linear Regression on basis of the RMSE scores. The same result is reinforced in [7] where multiple classification approaches are tried for prediction of Parkinson's and the deep neural network seems to perform the best. The performance of the classifiers was judged on the basis of the results of the confusion matrix of these algorithms. The work done in [8] makes clusters using Expectation Maximization and then applies Principal Component Analysis to the clusters obtained to address multicollinearity problems. These features are then trained on Support Vector Regression and Adaptive Neuro-Fuzzy Inference System models to provide regressive values of the Parkinson Disease progression (which is UPDRS scale), bolstering the idea of using clusters to identify the alike data points and then provide a regression or a classifying analysis to it. It was claimed by the authors that it was the

first time that dimensionality reduction and clustering method with EM was used in the context of Parkinson Diagnosis.

In [9], there was a discussion and comparative analysis on the methodology of selection of K-value in K-Means Clustering using four different K-value selection algorithms, which were the Silhouette Coefficient, Canopy algorithm, Gap Statistic and the Elbow method. The paper concludes that all four methods are suitable when performing clustering for small datasets. The study done in [10], [11] discusses the high potential of the usage of Gaussian Mixture Clustering models in multidimensional data on speech features. The algorithm is tested on varying multifaceted data derived from the sales of a global video game. The results of the experiments highlight the strengths and limitations of the Gaussian mixture model for clustering relatively large data. The methods described in [12] give a well laid foundation on deciding the right total optimum clusters using Hierarchical Clustering on different datasets, such as convex, manifold, linear and annular structures using a novel clustering validity index approach. There were specific guidelines demonstrated for the Eps Parameter Estimation of DBSCAN Clustering Algorithm in [13]. It provides an improvement on the self-adaptive selection of parameters. The result is far better clustering results at the cost of computational complexity.

To find the total optimum clusters using any clustering technique, multiple metrics or scores can be used in conjunction with them. In [14], the criterion of silhouette width is discussed for deciding the optimum clusters in the context of selecting image features from satellite forest image data in Japan. It concludes that the usage of this metric helps select a discriminative set of features hence leading to a highly accurate classification model. Another metric called Calinski-Harabasz score works on dispersion degree between the clusters is discussed in [15] where it is used along with silhouette score to develop a new improved metric based on both of them called Peak Weight Index. Calinski-Harabasz index proves to be important as it demonstrates the ability to improve the variations in results for clustering in the dataset. Davies-Bouldin metric for evaluation of clustering performance is introduced in [16] that can be integrated with a clustering algorithm to find the optimal number of clusters with low cost. The Davies-Bouldin index-based hierarchical initialization K-means algorithm proposed in the paper achieves on Gaussian distributions that are extremely normal in real-world use cases and thus demonstrates its effectiveness.

Our proposed methodology builds on this literature and improves upon it to build a more robust and reliable means of diagnosing Parkinson's and identifying its severity levels.

3 Proposed Work

The dataset has been extracted from [1], published in UCI ML repository. After performing some essential preprocessing operations, using four clustering algorithms K-Means, Agglomerative Hierarchical, Gaussian Mixture and DBSCAN the total optimum clusters are discovered. Using metrics like Calinski-Harabasz, Silhouette and Davies Bouldin Score Elbows help to identify the total optimum clusters according to each algorithm and by making use of a comparative analysis the optimal clusters for these data points is suggested to be four. After considering Silhouette, Calinski Harabasz and Davies Bouldin Scores on each algorithm K-Means, Agglomerative Hierarchical, Gaussian Mixture and DBSCAN with 4 clusters, the chief algorithm is used (Gaussian Mixture in this case) to generate clusters for the data points. Using t-SNE these clusters could be expressed visually in a two-dimensional representation. A multilayer stacked neural network is then constructed, with the preprocessed dataset at the input layer and the four severity layers set at the output layer. There are three hidden layers considered, given the dimensionality and the procedures adopted in [4]. This system then acts as a classifier for the four severity levels- “Level 1”, “Level 2”, “Level 3” and “Level 4”. After normalizing, data is used for training and testing of the modeled neural network.

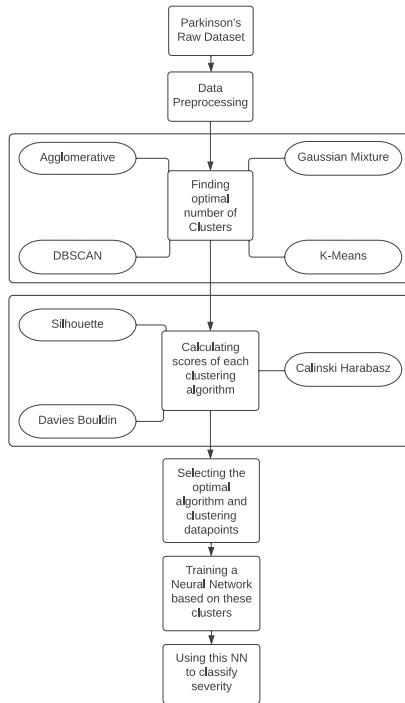


Fig 1. The system's proposed Methodology has been outlined in the form of a flow chart.

3.1 Data Collection

The Parkinson’s Disease: Oxford Telemonitoring Dataset from UCI ML Repository has been used to build our model. This consists of the collection of biomedical voice data points for nearly 42 Parkinson’s patients in early stages that were monitored remotely for symptom progression. This contains data points having 22 features including the subject number, subject age, subject gender, time interval (in days) from subjects’ recruitment, motor UPDRS, total UPDRS and 16 different biomedical voice attributes. There are 5875 instances in total, each corresponding to a voice recording from these patients, and there are around 200 recordings per patient. The dataset is in ASCII comma-separated values (CSV) format. The attributes used to build this dataset are shown in Table 1.

Features	Description of the Features
subject#	Identification Number for each test subject
age	Age of the Subject
sex	Gender of the Subject where '0' stands for male, '1' stands for female
test_time	The number of days passed from the recruitment day to the trial.
motor_UPDRS	Motor UPDRS score by the Clinician (linearly interpolated)
total_UPDRS	Total UPDRS score by the Clinician (linearly interpolated)
Jitter(%), Jitter(Abs), Jitter:RAP, Jitter:PPQ5, Jitter:DDP	Various attributes for measuring variation happening in the fundamental frequency. RAP: Relative Average Perturbation, the average absolute difference between a period and the average of it and its two neighbours; PPQ: five-point Period Perturbation Quotient; DDP: Average absolute difference of differences between cycles, divided by the average period
Shimmer, Shimmer(dB), Shimmer:APQ3, Shimmer:APQ5, Shimmer:APQ11, Shimmer:DDA	Various attributes for measuring variation happening in the amplitude. APQ: Three, Five, Eleven point Amplitude Perturbation Quotients; DDA: Average absolute difference between consecutive differences between the amplitudes of consecutive periods
NHR, HNR	Harmonics-to-Noise and Noise-to-Harmonics Ratio for the tonal components measured in voice
RPDE	Recurrence Period Density Entropy is a complexity measure that is dynamic
DFA	Detrended Fluctuation Analysis exponent in signal waves
PPE	Pitch Period Entropy for finding variation in Fundamental frequency

Table 1: Attributes used in the Parkinson’s Disease: Oxford Telemonitoring Dataset

3.2 Data Preprocessing

Using Pandas library to drop the data points having missing values and making use of Principal Component Selection for selecting the relevant attributes of the dataset. The irrelevant columns like the ‘subject#’ and ‘test_time’ are filtered out, leaving 20 attributes to train on. The data is considered for normalization, changing the shape of distribution of the multidimensional data to eliminate redundancy for efficient clustering methods.

3.3 Clustering Techniques

Four different clustering techniques have been employed to estimate the optimal number of clusters, namely K-Means Clustering, Agglomerative Clustering, DBSCAN and Gaussian Mixture.

K-Means Clustering

It's an unsupervised algorithm that learns to cluster a dataset into k partitions. It works on an algorithm that is centroid based, where each cluster is associated with a corresponding centroid and the objective is to reduce the sum of distances between points and its belonging clusters. Based on the Python implementation these clusters could vary based on the initialization, random state and algorithm to be followed ("lloyd", "elkan", "auto", "full").

Agglomerative Clustering

Also known as hierarchical agglomerative clustering, has the feature of not pre specifying the total clusters. It works by considering a singleton cluster and then repeatedly agglomerating pairs of clusters to the point that all clusters are combined into a single large cluster containing the complete data. There are several parameters to define them like affinity, linkage.

DBSCAN

Stands for Density Based Spatial Clustering of Applications with Noise is an algorithm that clusters by identifying dense regions by grouping together data elements that are measured based on distance to be close to each other. This focuses in each cluster point having the least number of points in its neighborhood for a given radius. The number of clusters obtained depend on the minimum samples taken for the model, showing its dependence on metrics and 'eps' epsilon value identified using the Knee Locator method.

Gaussian Mixture

The Gaussian Mixture model is a distribution-based clustering algorithm. A probabilistic model that expects all data points are produced from a combination of a limited number of Gaussian distributions with unknown variables. The probability for each data point in a set of data points belonging to each of the distributions is identified. These models vary based on covariance types ("spherical", "tied", "diag", "full") and random state considered while training it on Python.

To evaluate the optimal clusters suggested by each of these algorithms, this work makes use of the Silhouette, Calinski Harabasz and Davies Bouldin Scores which provide elbow on the optimal clusters to be used and respective scores for using these optimal clusters. The theoretical implications and reasoning of these statistical scores could be expressed as follows.

Silhouette Coefficient

A metric to calculate the goodness of a clustering model. It has an upper limit from 1 to the lower limit of -1, and its calculation for each sample is based on the means of intra-cluster distance, nearest-cluster distance. For a point i the Silhouette Coefficient is as (1).

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (1)$$

where, b(i) represents the smallest average of distances from point i to all points in any other cluster and a(i) represents the average of distances from point i to all other points in its own cluster. The Silhouette Coefficient identifies the correct assigning for individual points to their clusters.

- Having S(i) nearer to 0 means the point is between two clusters.
- Having it nearer to -1, implies to assign it to the other clusters for better accuracy.
- Having S(i) nearer to 1, implies the point belonging to the ‘correct’ cluster.
So, in a given dataset, the score with the local maximum Silhouette Coefficient is considered to have the optimal clusters.

Calinski Harabasz Index

According to this index good clusters are those which are well spaced from each other. By dividing the variation of the distances of individual objects to their cluster center by the sum of squares of distances between cluster centers, its calculated. Having a higher Calinski-Harabasz Index value, results in better clustering. Calinski-Harabasz Index is formulated as (2).

$$CH_k = \frac{BCSM}{k-1} \cdot \frac{n-k}{WCSM} \quad (2)$$

here, k is the number of clusters, n is the number of records in data, BCSM (between cluster scatter matrix) calculates separation between clusters and WCSM (within cluster scatter matrix) calculates compactness within clusters.

Davies Bouldin Index

The Davies-Bouldin index is used for assessing clustering algorithms. It is an internal assessment scheme, where the wellness of the clustering has been done using quantities and attributes implicit to the dataset. It's determined by the average likeness of each cluster with a most similar cluster to it. Having lower average likeness is, results in better clustering. The formula for Davies-Bouldin index is (3).

$$DB \equiv \frac{1}{N} \sum_{i=1}^N D_i \quad (3)$$

Where N represents the total cluster number,

$$D_i \equiv \max_{j \neq i} R_{i,j} \quad (4)$$

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}} \quad (5)$$

Where S_i represents the intra-cluster dispersion for cluster i,

S_j represents the intra-cluster dispersion for cluster j,

M_{ij} represents the distance between centroids for clusters i and j,

These scores can be calculated using the `sklearn.metrics` library in Python for finding values for each cluster. Once the number of clusters proposed by each algorithm are considered for voting, a consensus for the optimal clusters is gained. Using the optimal clusters with the parameters used in testing, the most optimal clustering algorithms are produced which are then again compared on these scores to get the most optimal clustering algorithm, with the resultant clusters.

3.4 Classifier Neural Network

These clusters are then trained on our neural network classifier built using Keras. The model contains 20 features for the input layer, 10 neurons are set for the first hidden layer, 20 neurons are set for the second hidden layer, 10 neurons are set for the third hidden layer, and finally the output layer is set to 4 neurons for each level of severity. The split ratio used on training and testing sets is 80:20. The input along with the hidden layers use the activation function Rectified Linear Unit (ReLU) that outputs directly if it is positive, and zero otherwise. The range of this function is thus 0 to positive infinity. The activation function Softmax is used for the output layer, as it is useful for multi-class classification problems where there are more than two class labels. The range of this function is 0 to 1. Overall, its compiled on the optimizer Adam and the loss function Categorical Cross Entropy, training it for 15 epochs and a batch size of five then assessing against its accuracy, to judge the reliability of the built classifier. Once trained, the model is ready to classify input data into any of the four possible severity levels of Parkinson's disease.

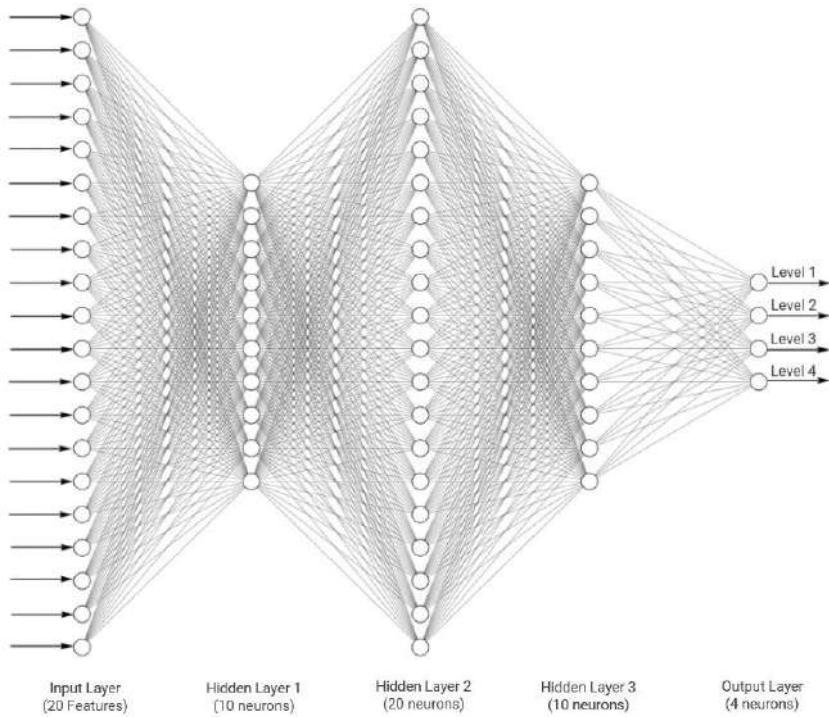


Fig 2. Multilayered Neural Network Classifier Architecture Diagram

4 Results Analysis

Python 3.10 was used to simulate the results of our experiments. From the dataset parameters discussed in the data collection section, the columns irrelevant to our work like the ‘subject#’ and ‘test_time’ were filtered out before moving on to the clustering of the individual data points. In this paper, four kinds of clustering algorithms K-Means, Agglomerative, DBSCAN and Gaussian Mixture model clustering alongside evaluation metrics such as Silhouette score, Calinski-Harabasz score and Davies-Bouldin score are used for this experiment. The optimum clusters obtained using each of these clustering techniques with each scoring metric have been shown in Fig 3,4,5. KEElbowVisualizer, matplotlib and pyplot have been used to visualize these graphs.

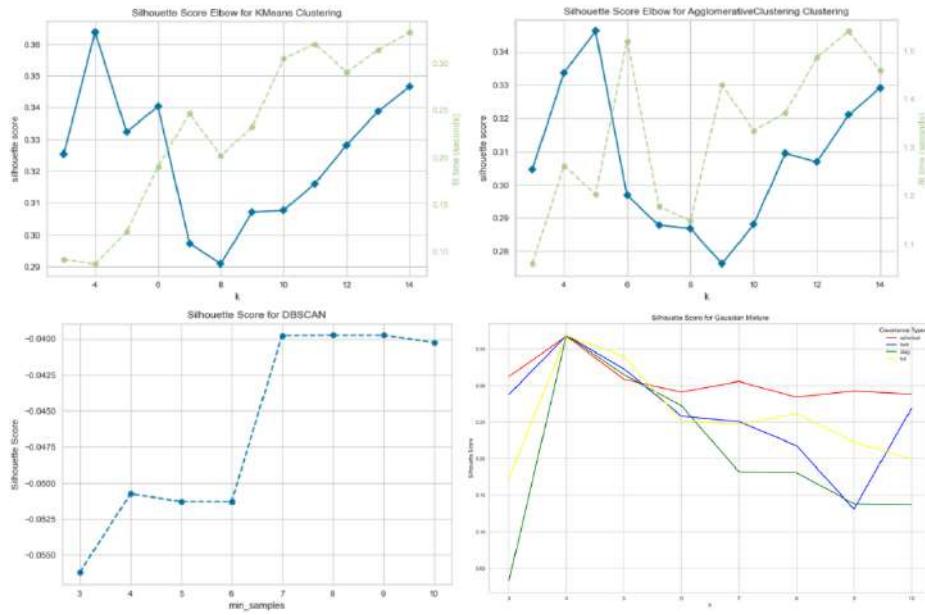


Fig 3. Plots of Silhouette Scores vs. No. of clusters for different clustering algorithms

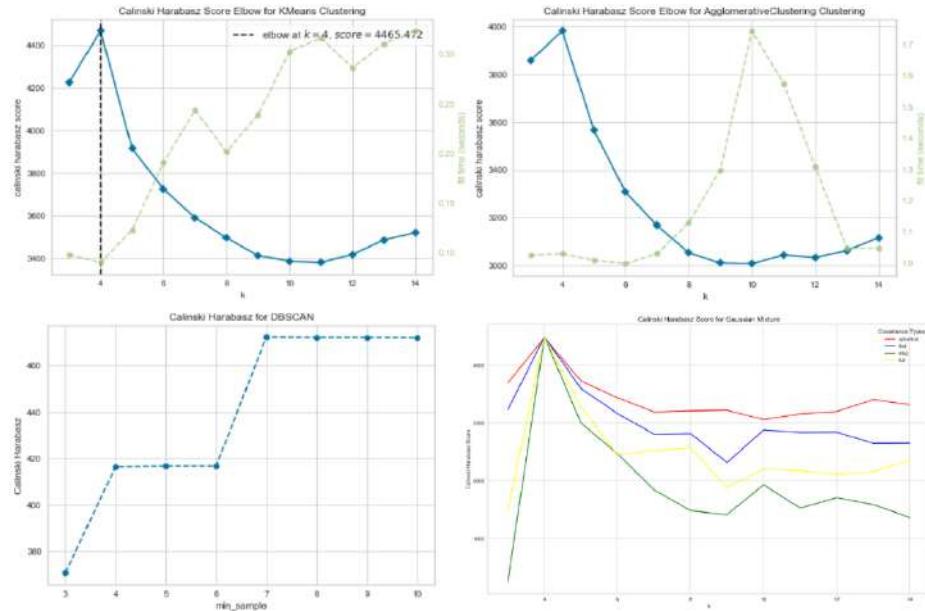


Fig 4. Plots of Calinski-Harabasz Scores vs. No. of clusters for different clustering algorithms

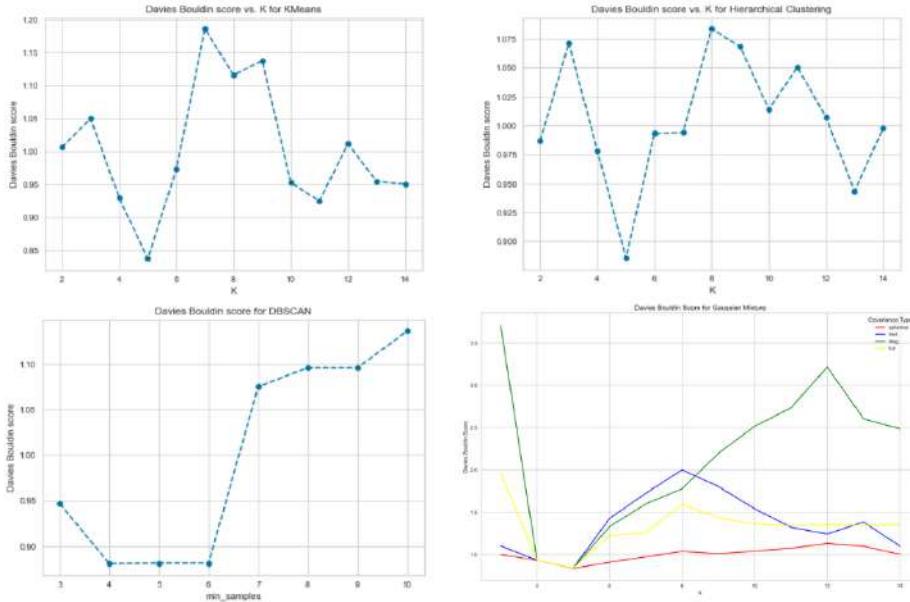


Fig 5. Plots of Davies Bouldin Scores vs. No. of clusters for different clustering algorithms

These representations are tabulated in Table 2.

S. No.	Clustering Technique	Optimum Clusters based on:		
		Silhouette Score	Calinski Harabasz Score	Davies Bouldin Score
1.	K-Means	4	4	4
2.	Agglomerative	5	4	5
3.	DBSCAN	8, 9	7	4
4.	Gaussian Mixture	4	4	8

Table 2: Clustering Techniques and Optimal Clusters advised by each score.

Based on the values in Table 2, it is observed that K-Means clustering results in 4 clusters based on all the scores, Agglomerative clustering results in 4 or 5 clusters, DBSCAN results in different no. of clusters for each metric used ranging from 4 to 9, and finally Gaussian Mixture model results in 4 or 8 clusters. Based on the

observations, 4 seems to be the most logical choice for the number of clusters. To be able to decide which clustering technique is to be relied upon for clustering into 4 clusters, we need to compare each of these techniques based on their scores for 4 clusters.

To decide the best clustering technique for 4 clusters, we calculate the Silhouette, Calinski-Harabasz as well as the Davies-Bouldin score for four clusters with each of the four clustering techniques and compare them here using Table 3.

S. No.	Clustering Technique	For 4 clusters		
		Silhouette Score	Calinski Harabasz Score	Davies Bouldin Score
1.	K-Means	0.3636	4465.4721	0.9385
2.	Agglomerative	0.3462	3566.2552	0.8998
3.	DBSCAN	-0.0372	474.2215	1.0734
4.	Gaussian Mixture	0.3675	4479.8375	0.9296

Table 3: Clustering Techniques and Scores for four clusters.

For deciding the optimal clustering technique, the one with a higher Silhouette score is deemed to be a better performing algorithm, its range being from -1 to +1. Similarly, a higher Calinski-Harabasz score is desirable when choosing a technique. On the contrary, a lower Davies-Bouldin score corresponds to algorithms with good clustering performance, with its minimum being 0. Based on these factors, the optimal clustering technique should be one with a higher Silhouette score, a higher Calinski-Harabasz score and a lower Davies-Bouldin score. The Gaussian Mixture model satisfies these conditions the best, and thus, should be the best clustering technique for our purpose.

Using T-distributed neighbour embedding (t-SNE), a dimensionality reduction technique that helps visualize high-dimensional data sets. The original data is entered into an algorithm and the best match representation using fewer dimensions of cluster data is generated. In figure 6 using 2 dimensions TC1(a component of the components) and TC2, this high dimensional data is represented with a perplexity (parametric term used in t-SNE) of 150.

Visualizing Clusters

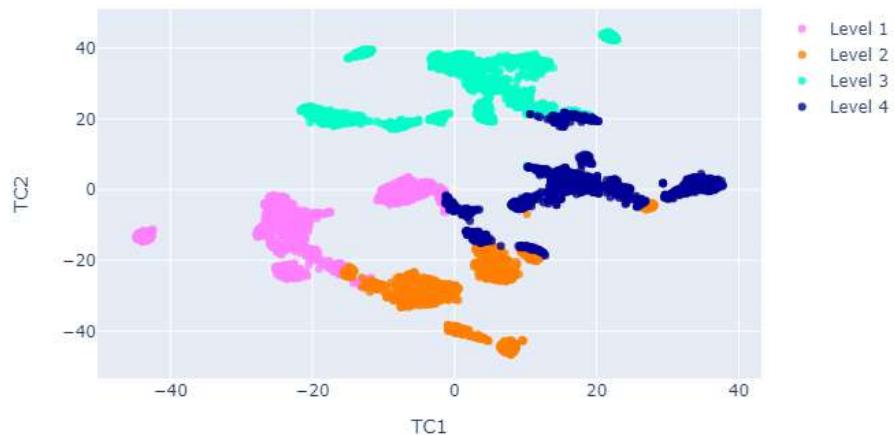


Fig 6. Cluster Visualization using t-SNE.

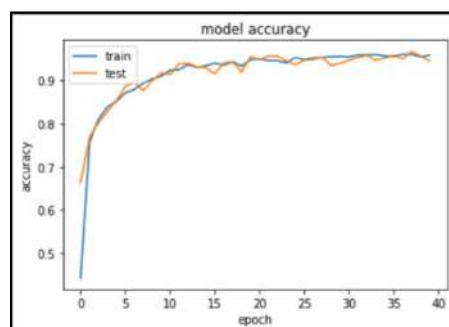


Fig 7. Epoch vs. Model Accuracy for Train and Test sets

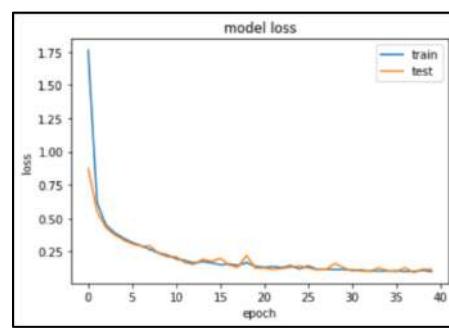


Fig 8. Epoch vs. Model Loss for Train and Test sets

Based on figure 7,8, it is inferred that using 0.01 as learning rate for the constructed neural network provides minimal overfitting. There is 95.8% accuracy and 0.1001 loss of the training data by the end of 15 epochs and 94.4% accuracy and 0.1146 loss of the testing data.

5 Conclusion

In this paper, we have implemented various clustering algorithms and used various scoring techniques to calculate the optimal number of severity levels of Parkinson's disease needed for a telemonitoring voice data. A deep neural network has been implemented to classify these data points into levels providing a severity rating for Parkinson's disease. These ratings have an indirect relation to the parameters, without any implied observation with the attributes individually. The accuracy of the constructed neural network is 94.4% with a loss of 0.1146. This is a good accuracy for our purpose. Thus, using the constructed neural networks, a data point could be ranked into a severity level. Although the dataset has 5875 instances, the precision of our methodology could be additionally improved by implementing it on a bigger dataset, having more cases of each severity class in combination with a data set of patients' voice information and supplementary patient highlights like handwriting and gait attributes.

References

1. Tsanas, Athanasios, Max Little, Patrick McSharry, and Lorraine Ramig. "Accurate telemonitoring of Parkinson's disease progression by non-invasive speech tests." *Nature Precedings* (2009): 1-1.
2. Martínez-Martín, Pablo, Carmen Rodríguez-Blázquez, Mario Alvarez, Tomoko Arakaki, Víctor Campos Arillo, Pedro Chaná, William Fernández et al. "Parkinson's disease severity levels and MDS-Unified Parkinson's Disease Rating Scale." *Parkinsonism & related disorders* 21, no. 1 (2015): 50-54.
3. Cai, T. Tony, and Rong Ma. "Theoretical Foundations of t-SNE for Visualizing High-Dimensional Clustered Data." *arXiv preprint arXiv:2105.07536* (2021).
4. Grover, Srishti, Saloni Bhartia, Abhilasha Yadav, and K. R. Seeja. "Predicting severity of Parkinson's disease using deep learning." *Procedia computer science* 132 (2018): 1788-1794.
5. Galaz, Zoltan, Zdenek Mzourek, Jiri Mekyska, Zdenek Smekal, Tomas Kiska, Irena Rektorova, Juan Rafael Orozco-Arroyave, and Khalid Daoudi. "Degree of Parkinson's disease severity estimation based on speech signal processing." In 2016 39th International Conference on Telecommunications and Signal Processing (TSP), pp. 503-506. IEEE, 2016.
6. Varghese, Basil K., D. Amali, and K. S. Devi. "Prediction of parkinson's disease using machine learning techniques on speech dataset." *Research Journal of Pharmacy and Technology* 12, no. 2 (2019): 644-648.
7. Haq, Amin Ul, Jianping Li, Muhammad Hammad Memon, Jalaluddin Khan, Salah Ud Din, Ijaz Ahad, Ruinan Sun, and Zhilong Lai. "Comparative analysis of the classification performance of machine learning classifiers and deep neural network classifier for prediction of Parkinson

- disease." In 2018 15th International computer conference on wavelet active media technology and information processing (ICCWAMTIP), pp. 101-106. IEEE, 2018.
- 8. Nilashi, Mehrbakhsh, Othman Ibrahim, and Ali Ahani. "Accuracy improvement for predicting Parkinson's disease progression." *Scientific reports* 6, no. 1 (2016): 1-18
 - 9. Yuan, Chunhui, and Haitao Yang. "Research on K-value selection method of K-means clustering algorithm." *J 2*, no. 2 (2019): 226-235
 - 10. Ahmed, Saadaldeen Rashid Ahmed, Israa Al Barazanchi, Zahraa A. Jaaz, and Haider Rasheed Abdulshaheed. "Clustering algorithms subjected to K-mean and gaussian mixture model on multidimensional data set." *Periodicals of Engineering and Natural Sciences (PEN)* 7, no. 2 (2019): 448-457.
 - 11. Kuresan, Harisudha, Sam Masunda, and Dhanalakshmi Samiappan. "Analysis of Jitter and Shimmer for Parkinson's disease diagnosis using telehealth." In *Cognitive Informatics and Soft Computing*, pp. 711-721. Springer, Singapore, 2019.
 - 12. Zhou, Shibing, Zhenyuan Xu, and Fei Liu. "Method for determining the optimal number of clusters based on agglomerative hierarchical clustering." *IEEE transactions on neural networks and learning systems* 28, no. 12 (2016): 3007-3017.
 - 13. Zhu, Lu, Jiang Zhu, Chongming Bao, Lihua Zhou, Chongyun Wang, and Bing Kong. "Improvement of DBSCAN Algorithm Based on Adaptive Eps Parameter Estimation." In *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 1-7. 2018.
 - 14. Kaoungku, Nuntawut, Keerachart Suksut, Ratiporn Chanklan, Kittisak Kerdprasop, and Nittaya Kerdprasop. "The silhouette width criterion for clustering and association mining to select image features." *International journal of machine learning and computing* 8, no. 1 (2018): 69-73.
 - 15. Wang, Xu, and Yusheng Xu. "An improved index for clustering validation based on the Silhouette index and Calinski-Harabasz index." In *IOP Conference Series: Materials Science and Engineering*, vol. 569, no. 5, p. 052024. IOP Publishing, 2019.
 - 16. Xiao, Junwei, Jianfeng Lu, and Xiangyu Li. "Davies Bouldin Index based hierarchical initialization K-means." *Intelligent Data Analysis* 21, no. 6 (2017): 1327-1338.

Attendance Management System Using Face Recognition

I.G.Vishnu Vardhan¹, M.Ajay Kumar², P.Nithin Kalyan³, V.Spandana⁴,
Dr.J.Ebenezer^{*5},

^{1,2,3} Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada,
Andhra Pradesh, India.

⁴ Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, Andhra
Pradesh, India. ORCID: 0000-0001-8544-6930. kalyanichandrak@gmail.com^{*}

Abstract.

Today's instructional establishments are involved approximately students' constant performance. The inadequate attendance is one issue contributing to the decline in student performance. The maximum famous strategies to document your attendance are to signal or call the pupils. It became difficult and took longer. A computer-primarily based totally student attendance tracking gadget that permits the trainer to hold attendance records is now essential. In this project, we used an wise attendance system based on face recognition. We have counseled installing vicinity a "Attendance Management System Using Face Recognition" that has several uses. Due to face authorization, the contemporary implementation consists of facial identification, which saves time and gets rid of the opportunity of proxy attendance. Additionally, this determines whilst someone entered a selected established order and information their attendance. Python is used to put into effect this project, and Pycharm is used for development. In our work, the Face Landmark Estimation set of rules is employed. The challenge is to discover their faces and mark the attendance. Because the face is employed as a biometric for authentication, this technique gets rid of the opportunity of fraudulent attendance.

Keywords: Authorization, Face recognition, Identification, OpenCV, dlib, cv2, os and some python modules.

1 Introduction

This shows the total overview and outline of our project the necessity to ensure the security of data or physical assets is become both more crucial and more challenging in todays networked society we occasionally learn about crimes like credit card fraud hacker attacks on computers or security lapses in a business or governmental structure the majority of these crimes were committed by people taking advantage of a fundamental shortcoming in traditional access control systems which provide access based on what we have such as id cards keys passwords pin numbers or mothers maiden name rather than who we are none of these methods truly define who we are technology has recently made it possible to verify a persons actual

identity the science behind this technology is known as biometrics biometric access control refers to automated techniques for confirming or detecting the identification of a living person based on physical traits like fingerprints or facial features or behavioral traits like a persons handwriting or keyboard patterns biometric systems are challenging to fake since they use biological traits to identify a person one of the few biometric techniques that has the advantages of high accuracy and minimal intrusion is face recognition it is accurate yet not obtrusive like a physiological approach for this reason face recognition has attracted the interest of academics in a variety of domains including security psychology image processing and computer vision since the early 1970s kelly 1970 2 today attendance is the most crucial factor to record a persons existence for every organization a persons attendance at an agency or organization is evidence that they are fulfilling their commitment to do so typically taking attendance is done by hand it can be yelled out individually or signed in order to speed and give time efficiency in this digital age there must be a change from this lack face recognition technology allows us to keep track of attendance for every person in a room in this face recognition a variety of techniques including machine learning are used to analyze and capture photos of someones face with the help of this algorithm the system can identify a person and record their attendance making the process more quick and effective

2 Review of Literature

In [1], Face features, or face implementation, are employed in this study as biometrics. This system suggested a method utilising OpenCV that combines a camera that takes a picture of the input, an algorithm for recognising faces in inputs, encoding & identifying the faces, and recording the attendance. This article uses the LBPH (Local Binary Pattern Histogram) and PCA algorithms (Principal Component Analysis). The LBPH approach is used for feature extraction. Additionally, it makes use of the PCA method, which is based on the Eigen face technique and compares the test and training images to identify who is present and who is not. When there are many students in a lecture, this technique saves time and effort. An Excel spreadsheet is used to track attendance.

In [2] The system in this paper makes use of Fisher face and Eigen face approaches. The PCA-based Eigen faces technique is utilised to reduce the dimensionality of the images. The LDA approach is the basis for Fisher face. Due to LBP's resilience to position and illumination changes, it was chosen. The difference between multiple faces is quantified using these Eigenvectors. After achieving positive results from numerous experimental examinations of this technique, this system has dependable results for posture variance, lighting, and processing the entire image. It also processes the entire image in a shorter amount of time.

In [3] Principal Component Analysis (PCA) was used to create this real-time system, and the research also compares PCA and Linear Discriminant Analysis (LDA) as facial recognition algorithms using the Olivetti dataset. First, the dataset is transformed 8 into a numpy array, and then it is saved in csv format. This system suggests employing feature-based approaches for facial recognition. These techniques

examine the geometric relationships between individual facial characteristics. To build these approaches, statistical tools like SVM, LDA, and PCA are employed.

In [4] The Open CV-based face recognition approach has been put forth in this work. This model incorporates a camera that takes an input image, a face detection method, encoding and face identification, registering attendance in a spreadsheet, and turning the spreadsheet into a PDF file. By teaching the system with the faces of the permitted pupils, the training database is established. Afterward, a database containing the cropped photographs is created and labelled accordingly. The LBPH algorithm is used to extract the features. The results of this paper may recognize images in different angle & light conditions. The recognized images of students is marked in real time and import to excel sheet.

3 Proposed Architecture and Methodology

3.1 Proposed Architecture

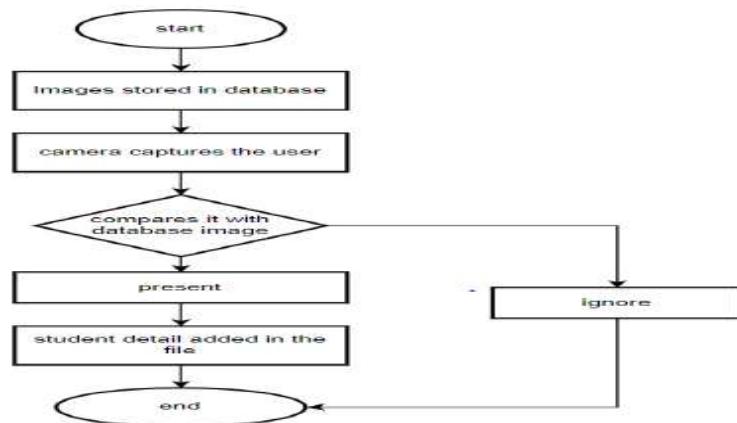


Fig 3.1: Architecture of the Figure methodology used for Face detection and Recognition

3.2 Proposed Methodology of the Work

3.2.1 Data Preprocessing

Data preprocessing is a crucial step in our project. In this step, we can store images of students with their names or registration numbers in our system. that we named the dataset. After this, by using the face-recognition module, the faces in front of the camera are detected, and after that, each face can be divided into sixty-eight measurements by using the Face Landmark Estimation algorithm; if any of the faces

are matched with faces that are stored in our dataset, then that student gets attendance.

3.2.2 Algorithm

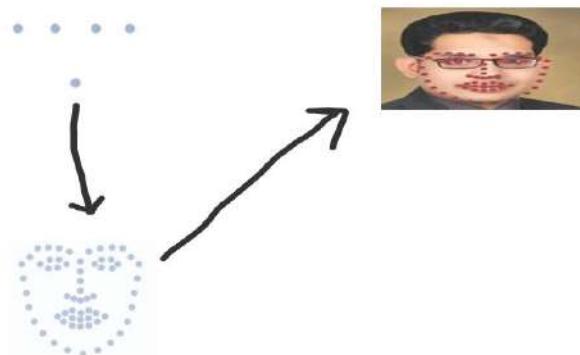
The following is the algorithm used for Attendance Management System Using Face Recognition Project.

Algorithm: Face Landmark Estimation Algorithm.

After detecting a face in an image, the Face landmark estimation set of rules identifies key factors at the detected face, which includes the end of the nostril and the centre of the eye. After this, the recognized face might be in comparison with the face this is saved in our scholar database, and if each the faces are matched, then that scholar receives attendance.



Figure 3.3: Working procedure of Face Landmark Estimation algorithm



Description of Datasets, Requirements and tools:

4.1 Dataset

The dataset is very important to our project. it entails gathering student images and storing them in our database. after that, verification of detected images and stored images should be done.Finally dataset means it contains images of students.



4.2.Pycharm

Pycharm is a source-code editor made with the green framework for windows linux and mac os features encompass help for debugging syntax highlighting intelligent code completion snippets code refactoring and embedded git users can change the theme keyboard shortcuts preferences and install extensions that add additional functionality pycharm is a code editor like many different code editors pycharm adopts an unusual place user interface and format of an explorer at the left displaying all the documents and folders you've get entry to and an editor at the right displaying the contents of the documents you've got opened.

```
EpicsProject main.py
EpicsProject ImagesAttendance
EpicsProject ImagesBasic
EpicsProject Attendance.csv
EpicsProject main.py
AttendanceProject.py Attendance.csv

1 import cv2
2 import face_recognition
3
4 imgGeetha = face_recognition.load_image_file('ImagesBasic/Sameena.jpg.jpeg')
5 imgGeetha = cv2.cvtColor(imgGeetha, cv2.COLOR_BGR2RGB)
6
7 imgTest = face_recognition.load_image_file('ImagesBasic/Sameena Test.jpg.jpeg')
8 imgTest = cv2.cvtColor(imgTest, cv2.COLOR_BGR2RGB)
9 faceLoc = face_recognition.face_locations(imgGeetha)[0]
10 encodeGeetha = face_recognition.face_encodings(imgGeetha)[0]
11 cv2.rectangle(imgGeetha, (faceLoc[3], faceLoc[0]), (faceLoc[1], faceLoc[2]), (255, 0, 255), 2)
12 faceLocTest = face_recognition.face_locations(imgTest)[0]
13 encodeTest = face_recognition.face_encodings(imgTest)[0]
14 cv2.rectangle(imgTest, (faceLocTest[3], faceLocTest[0]), (faceLocTest[1], faceLocTest[2]), (255, 0, 255), 2)
15 results = face_recognition.compare_faces([encodeGeetha], encodeTest)
16 faceDis = face_recognition.face_distance([encodeGeetha], encodeTest)
17 print(results, faceDis)
18 cv2.putText(imgTest, f'{round(faceDis[0], 2)}', (50, 50), cv2.FONT_HERSHEY_COMPLEX, 1, (0, 0, 255), 2)
```

Python:

Machine learning may be an incredibly beneficial device to find hidden insights and expect destiny trends this machine learning with python direction will supply you all of the gear you want to get began out with advice systems python is a high level interpreted interactive and object oriented scripting language python designed to be highly readable it uses English keywords frequently whereas different languages use punctuation and it has fewer syntactical constructions than different languages.

Dlib:

Dlib is a cutting-edge, C toolkit that includes machine learning techniques and tools for developing sophisticated software to address real-world issues. It is employed in a variety of fields, including robots, embedded technology, mobile phones, and huge high-performance computer environments in both industry and academia. dlib can be used for free in any application thanks to its open source licensing.

Implementation Steps:

1. Collecting images of students and storing them in our database.
2. Install the libraries or packages in pycharm which are useful to our project.
3. And import all the packages that are installed in step2.
4. And write the python code for face detection, comparing, and for displaying time in excel sheet.
5. In this step, images of students could be captured by camera in appropriate sizes.

6. And then detected images will be compared with the stored images, if detected image will be matched with the stored image then goto step7 and step8.
7. If the captured student is present in our database then that student gets attendance
8. If not present just ignore that student.
9. If step7 is done successfully then details of that student will be displayed in the Excel sheet.
10. That's how implementation is done.

Result Analysis:

The overall window configuration of the attendance management system is shown below.

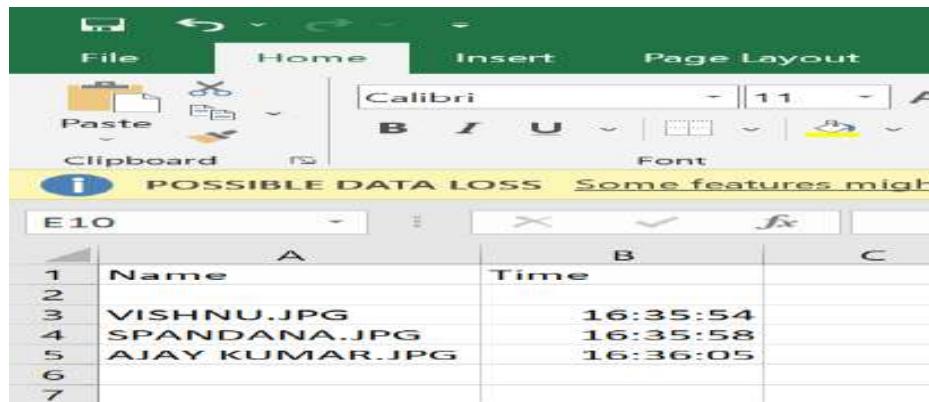
Face Recognition:

The input image is captured by the camera and matches to the features that are extracted earlier and detected the name and identity of the person.



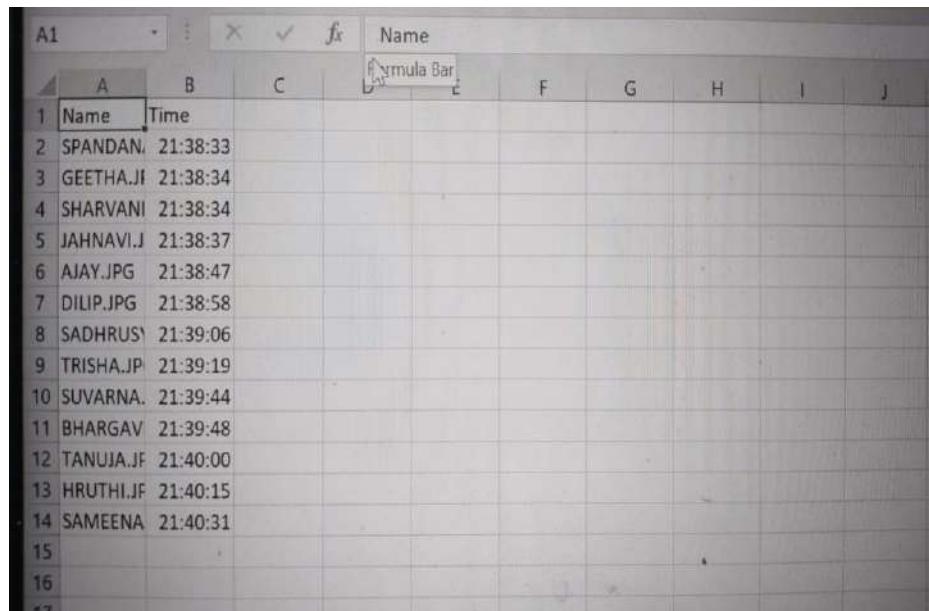
Marking Attendance in Excel sheet:

The attendance of the student got marked and recorded in the csv file with Name, Date and Time of the person entered.



A screenshot of Microsoft Excel showing a table of student attendance data. The table has two columns: 'Name' and 'Time'. The data is as follows:

	Name	Time
1	VISHNU.JPG	16:35:54
2	SPANDANA.JPG	16:35:58
3	AJAY KUMAR.JPG	16:36:05
4		
5		
6		
7		



A screenshot of Microsoft Excel showing a table of student attendance data. The table has two columns: 'Name' and 'Time'. The data is as follows:

	Name	Time
1	SPANDAN	21:38:33
2	GEETHA.JI	21:38:34
3	SHARVANI	21:38:34
4	JAHNAVI.J	21:38:37
5	AJAY.JPG	21:38:47
6	DILIP.JPG	21:38:58
7	SADHRUS	21:39:06
8	TRISHA.JP	21:39:19
9	SUVARNA.	21:39:44
10	BHARGAV	21:39:48
11	TANUJA.JF	21:40:00
12	HRUTHI.JF	21:40:15
13	SAMEENA	21:40:31
14		
15		
16		
17		

5 Conclusion:

Face Recognition based Attendance Management system was the project chosen by us by keeping in view of the demands of day to day needs and wants of the society. The advancements in technology lead us to think out of the box and come up with some idea that could be future changing. Education is the most important thing which every person should acquire as it is the basis for a better lifestyle and will surely alleviate the standard of a living community. What our education system lacks is the involvement of students in the schools, colleges and universities. Instead of attending lectures and studying they prefer staying away from class and keep engaged in using these gadgets. Low attendance means that the students are not there to acquire the knowledge which they are supposed to get and is of immense importance for them and can lead them to a better future.

References

- [1] N.Vishwanadha Reddy, K.Roshini,Prashant Mishra, Thulasi Thirumaleswari, and Durga S Chandhu. "Smart attendance system using face recognition." In 2021 A peer reviewed open access International Journal(IJRACSE), pp. 2454-4221,2021.
- [2] Preethi, Kolipaka, and Swathy Vodithala. "Automated Smart Attendance System Using Face Recognition." In 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 1552-1555. IEEE, 2021.
- [3] Karthick, S., S. Selvakumarasamy, C. Arun, and Pratik Agrawal. "Automatic attendance monitoring system using facial recognition through feature-based methods (PCA, LDA)." Materials Today: Proceedings (2021).
- [4] Bussa, Sudhir, Ananya Mani, Shruti Bharuka, and Sakshi Kaushik. "Smart attendance system using OPENCV based on facial recognition." Int. J. Eng. Res. Technol 9, no. 3 (2020): 54-59.
- [5] Raj, A. Arjun, Mahammed Shoheb, K. Arvind, and K. S. Chethan. "Face Recognition Based Smart Attendance System." In 2020 International Conference on Intelligent Engineering and Management (ICIEM), pp. 354-357. IEEE, 2020.
- [6] L. Li, X. Mu, S. Li and H. Peng, "A Review of Face Recognition Technology," in IEEE Access, vol. 8, pp. 139110-139120, 2020, doi: 10.1109/ACCESS.2020.3011028.

Diabetes classification using ML algorithms

G. G. Rajput¹ and Ashvini Alashetty ²

^{1,2}Department of Computer Science, Karnataka State Akkamahadevi Women's University,

Vijayapura, Karnataka-586108

¹ggrajput@yahoo.co.in, ²ashwinialashetty@gmail.com

Abstract. Healthcare organizations accumulate huge amount of data including electronic health records, images, omics data, and text but gaining knowledge and insight into the data remains a key challenge. The latest advances in Machine learning technologies can be applied for obtaining hidden patterns, which may diagnose diabetes at an early phase. This work presents a methodology for diabetes prediction is using a diverse machine learning algorithm such as SVC, Decision Tree Classifier, K-Neighbours Classifier, Logistic Regression, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier using the PIMA Indian dataset. Hence, this proposed system provides an effective prognostic tool for healthcare officials. The results obtained can be used to develop a novel automatic prognosis tool that can be helpful in early detection of the disease. The analysis on Pima Indian Diabetes Dataset (PIDD) is carried out by splitting dataset in to 90% training data and 10% testing data., the Support vector classifier achieved the best overall performance with 79% accuracy and 0.78 and 0.79 F1 score and precision respectively.

Keywords: Machine learning; Decision Tree Classifier; Diabetes prediction; PIMA Indian dataset.

1. Introduction

In India, diabetes is a major issue. Diabetes is a disorder that occurs either when the pancreas does not produce enough insulin or when the body cannot effectively use the insulin, it produces. Glucose is an important source of energy for the cells that make up the muscles and tissues. It is also the main source of fuel of the brain. The main cause of diabetes varies by type. But, no matter what type of diabetes you have, it can lead to excess sugar in the blood. Too much sugar in the blood can lead to serious health

problems. Chronic diabetes conditions include type 1 diabetes and type 2 diabetes. Potentially, reversible diabetes conditions include pre-diabetes and gestational diabetes. Pre-diabetes happens when blood sugar level is higher than normal. However, the blood sugar level could not be high enough to be called diabetes. Moreover, pre-diabetes can lead to diabetes unless steps are taken to prevent it. On the other hand, gestational diabetes happens during pregnancy. In India there are reportedly 77.2 million people with pre-diabetes. In 2012, nearly 1 million people in India died of diabetes. 1 out of 4 individuals living in Chennai's urban slums suffer from diabetes, which is about 7 per cent by three times the national average. One third of the deaths in India involve people under non-communicable diseases Sixty years old. Indians get diabetes 10 years before their Western counterparts on average. Changes in lifestyle lead to physical decreases increased fat, sugar and activities activity calories and higher insulin cortisol levels Obesity and vulnerability. In 2019, diabetes was the direct cause of 1.5 million deaths and 48% of all deaths due to diabetes occurred before the age of 70 years. Another 460000 kidney disease deaths were caused by diabetes, and raised blood glucose causes around 20% of cardiovascular deaths. In Early diagnosis, the prediction and diagnosis of the disease are analyzed through a doctor's knowledge and experience, but that can be inaccurate and susceptible. Healthcare Industry collects a huge amount of data related to healthcare, but that data is unable to perceive undetected patterns for making effective decisions. Since manual decisions can be highly dangerous for early disease diagnosis as they are based on the healthcare official's observations and judgment which is not always correct. There can be some patterns that remain hidden and can impact observations and outcomes. As a result, patients are getting a low quality of service; therefore, an advance mechanism is required for early detection of disease with an automated diagnosis and better accuracy. Various undetected errors and hidden patterns give rise to diverse data mining and machine algorithms which can draw efficient results with reliable accuracy. Due to the day to day growing impact of diabetes, a variety of data mining algorithms have been introduced for collecting hidden patterns from large healthcare data. Further, that data can be used for feature selection and automated prediction of diabetes. The main intent of this research work is to propose the development of a prognostic tool for early diabetes prediction and detection with improved accuracy. There have been an extensive amount of data and datasets available

on the internet or external sources and the PIMA Indian dataset which has been used in this work is one of the most widely used dataset in many researches and it is collected by the National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK). Pima Indian Diabetes Dataset consists of eight parameters. Those parameters include the number of times pregnancy has occurred, BMI, plasma glucose, diastolic blood pressure, systolic blood pressure, skinfold thickness, diabetic pedigree function, and Class 0 or 1 (0 means non-diabetic while 1 means diabetic patient). This research work represents comprehensive studies done on the PIMA Indian datasets using data mining algorithms like Decision Tree Classifier, K-Neighbours Classifier, Logistic Regression, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier. The combination of algorithms is represented in a logical and well-organized manner from which combination of both algorithms provides more effective and prominent results.

2. Literature Review

Deepti Sisodia [4]designed a model which can prognosticate the likelihood of diabetes in patients with maximum accuracy. Therefore, three machine learning classification algorithms namely Decision Tree, SVM and Naive Bayes are used in this experiment to detect diabetes at an early stage. Experiments are performed on Pima Indians Diabetes Database (PIDD) which is sourced from UCI machine learning repository. The performances of all the three algorithms are evaluated on various measures like Precision, Accuracy, F-Measure, and Recall. Accuracy is measured over correctly and incorrectly classified instances. Results obtained show Naive Bayes outperforms with the highest accuracy of 76.30% comparatively other algorithms. These results are verified using Receiver Operating Characteristic (ROC) curves in a proper and systematic manner. Francesco Mercaldo[5]demonstrated that diabetes pathology is increasing in last decades and the trend do not tends to stop. In order to help and to accelerate the diagnosis of diabetes in this paper we propose a method able to classify patients affected by diabetes using a set of characteristics selected in according to World Health Organization criteria. Evaluating real-world data using state of the art machine learning algorithms, we obtain a precision value equal to 0.770 and a recall equal to 0.775 using the HoeffdingTree algorithm.

FareehaAnwar[6] performed a literature review for diabetes diagnosis approaches using Artificial Intelligence (neural networks, machine learning, deep learning, hybrid methods, and/or stacked-integrated use of different machine learning algorithms). More than thirty-five papers have been shortlisted that focus on diabetes diagnosis approaches. Different datasets are available online for the diagnosis of diabetes. Pima Indian Diabetes Dataset (PIDD) is the most commonly used for diabetes prediction. In contrast with other datasets, it has key factors which play an important role in diabetes diagnosis. This survey also throws light on the weaknesses of the existing approaches that make them less appropriate for a diabetes diagnosis. In artificial intelligence techniques, deep learning is widespread and in medical research, heart rate is getting more attention. Deep learning combined with other algorithms can give better results in diabetes diagnosis and heart rate should be used for other cardiac disease diagnoses.

Huma Naz[7]Stated machine learning technologies can be applied for obtaining hidden patterns, which may diagnose diabetes at an early phase. This research paper presents a methodology for diabetes prediction using a diverse machine learning algorithm using the PIMA dataset. The accuracy achieved by functional classifiers Artificial Neural Network (ANN), Naive Bayes (NB), Decision Tree (DT) and Deep Learning (DL) lies within the range of 90-98%.

BalaManojKumarP[8] stated diabetes is a severe disease, most of the people are not aware of the risk associated with the disease because of that people die due to diabetic nephropathy, cardiac stroke and some other disorders. Therefore, early identification of diabetes helps to maintain sound health and life. Deep Learning approaches are used to predict diabetes accurately as humans do. In this paper, Deep Neural Network (DNN) classifier, an unsupervised learning approach is used for accurate prediction on Pima Indian Diabetes dataset and Feature Importance model that is bagged with Extra Trees and Random Forest is used for feature selection. The Pima Indian Diabetes dataset (PID) was acquired from the repository of UCI. The existing dataset has experimented with different formats of train test splits. The performance of the model was evaluated through accuracy, specificity, sensitivity, recall and precision. The model achieved 98.16% accuracy with random train-test split and it is observed that, the model obtained better performance than other state-of-art methods.

3. Proposed Methodology

This proposed methodology consists of two main parts, first how accuracy is obtained using diverse classification models and second is model validation. There are various machine learning methodologies available that are constructive to analyses the undetected patterns for evaluation of risk factors in diseases like diabetes. Further, it is being observed that the presentation of conventional methods is not up to the acceptance level in speech and object recognition because of a high dimension of data. A lot of research has been done in healthcare by implementing ML in anomaly detection. Related to diabetes prediction, our proposed model achieved the highest accuracy to date on the PIMA dataset i.e., 79.33%.

In view of the problem statement described in the introduction section, we propose a classification model with boosted accuracy to predict the diabetic patient. In this model, we have employed different classifiers like SVC, Decision Tree Classifier, K-Neighbours Classifier, Logistic Regression, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier are applied on the PIMA dataset for the evaluation of efficiency that is directly proportioned to the accurate decisions. Our proposed method has been chosen based on a task associated with the prophecy of diabetes disease.

3.1 Workflow Diagram

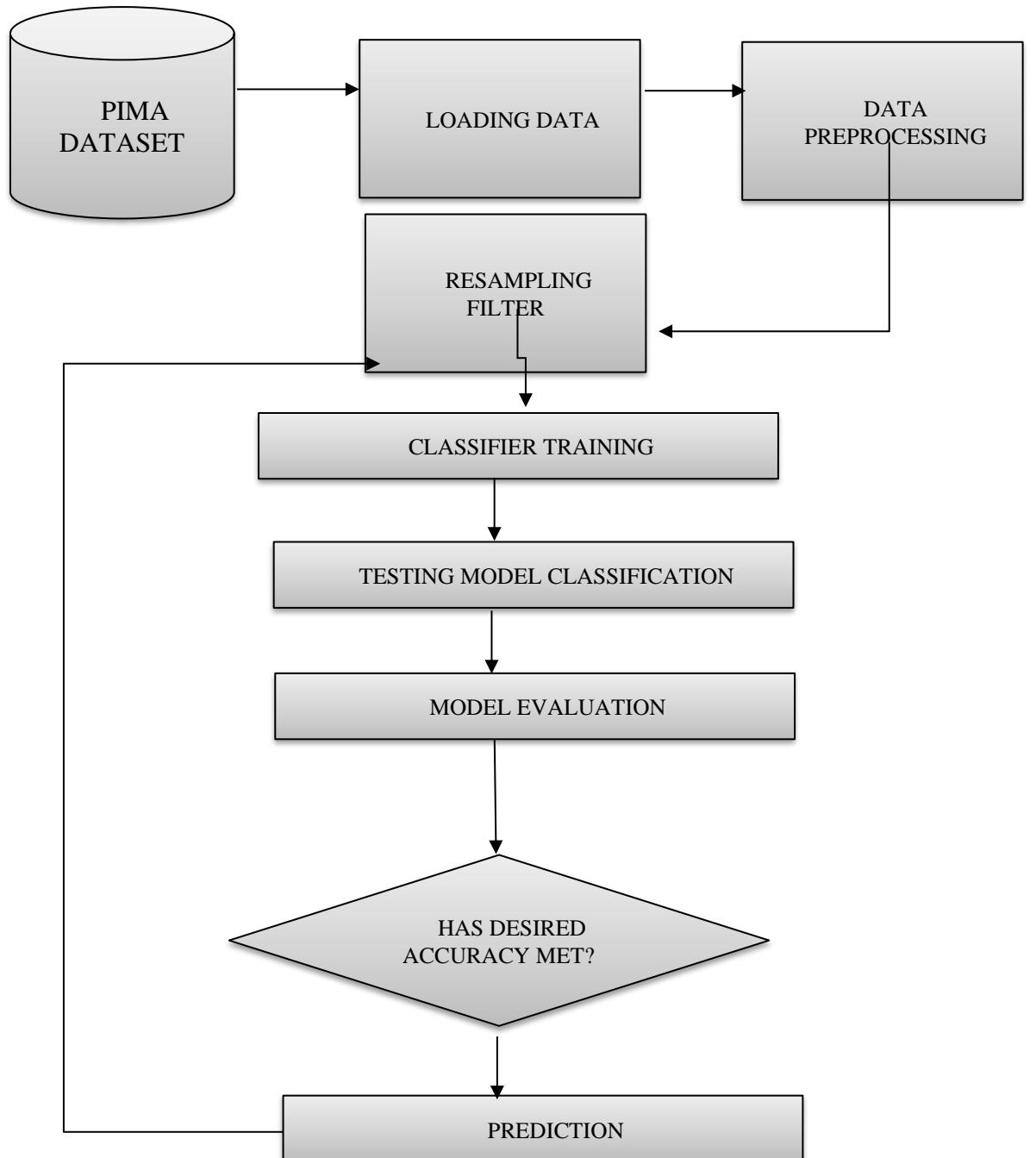


Figure 1: The workflow of the ML models

The framework is composed of the following important phases:

- Dataset Selection (PIMA Indian Diabetes Dataset)
- Data Pre-processing
- Resampling Filter
- Learning by Classifier (Training) i.e., Decision Tree Classifier, K-Neighbours Classifier, Logistic Regression, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier
- Achieving trained model with highest accuracy
- Using trained model for prediction

3.2 Dataset Selection

In data mining and machine learning, the data selection is a process in which the most relevant data is selected from a specific domain to derive values that are informative and facilitate learning within that domain. In the study, we have used diabetes dataset having eight attributes that are used to predict the symptom of gestational diabetes in a female patient. This dataset was obtained from UCI repository and is a benchmark dataset. On the basis of historical information stored in the dataset such as age, body mass index, blood pressure and number of times pregnant the classifiers are trained for making decision whether diabetes test for an individual is positive or negative. The PIMA diabetes dataset only represents the Indian national females who are at least 21 years old. All of the attributes are of numeric-valued continuous data type.

3.2.1 PIMA Indian Dataset (PID)

The Pima Indian dataset is an open-source dataset [6] that is publicly available for machine learning classification. The dataset used for the study is PIMA Indian dataset (PID) by NIDDK. The main motivation behind using the PIMA dataset is that most of the population in today's world follows a similar lifestyle having a higher dependency on processed foods with a decline in physical activity. PID is a long-term cohort study since 1965 by NIDDK because of the maximum risk of diabetes. The dataset contained

certain diagnostic parameters and measurement through which the patient can be identified with any kind of chronic disease or diabetes before time. All of the Participants in PID are females and at least 21 years old. PID composed of a total of 768 instances, from which 268samples were identified as diabetic and 500 were non-diabetics. The 8 most influencing attributes that contributed to-wards the prediction of diabetes are as follows: several pregnancies the patient has had, BMI, insulin level, age, Blood Pressure, Skin thickness, Glucose, Diabetes Pedigree Function with label outcome. Cross-validation has been used for estimating the statistical performance of the learning model. It executes two sub-processes as testing and training. The training subprocess is used to train a model and then the learning model is applied in the Testing subprocess to measure the accuracy. The reason for choosing Pima Indian dataset is the high prevalence of type 2 diabetes in the Pima group of Native Americans living in the area which is now known as central and southern Arizona. This group has survived with a poor diet of carbohydrates for years because of the genetic pre-disposition. In recent years, the Pima group gain a high indication of diabetes due to the sudden shift from traditional crops to processed foods.

Table1. PIMA Diabetes real-time Dataset.

S. No	Parameters	Description of parameters	Range
1	Pregnancies	No. of times pregnant	0–17
2	Glucose	Plasma glucose 2 h in an oral glucose tolerance test(mg/dl)	0–199
3	Blood-pressure	Diastolic blood pressure (mm Hg)	0–122
4	Skin Thickness	Skin fold thickness (mm)	0–99
5	Insulin	2-h serum insulin (mu U/mL)	0–846
6	BMI	Body mass index (weight in kg/(height in m) ²)	0–67.1
7	Diabetes-Pedigree	Diabetes pedigree function (weight in kg/(height in m) ²)	0.08–2.42
8	Age	Age (years)	21–81

3.2.2 Real time dataset availability and training

This section provides a comprehensive description of the real-time PIMA Indian dataset which consists of 768 female patients who were between the ages of 21 and 25 years. There are 268 diabetics among them, and the rest are healthier ones. The dataset consists of 8 vital parameters, and a complete overview of the dataset is given in Table 1, where we input parameters and their ranges given, such as the number of times a woman was pregnant and expressed in Figure 1.

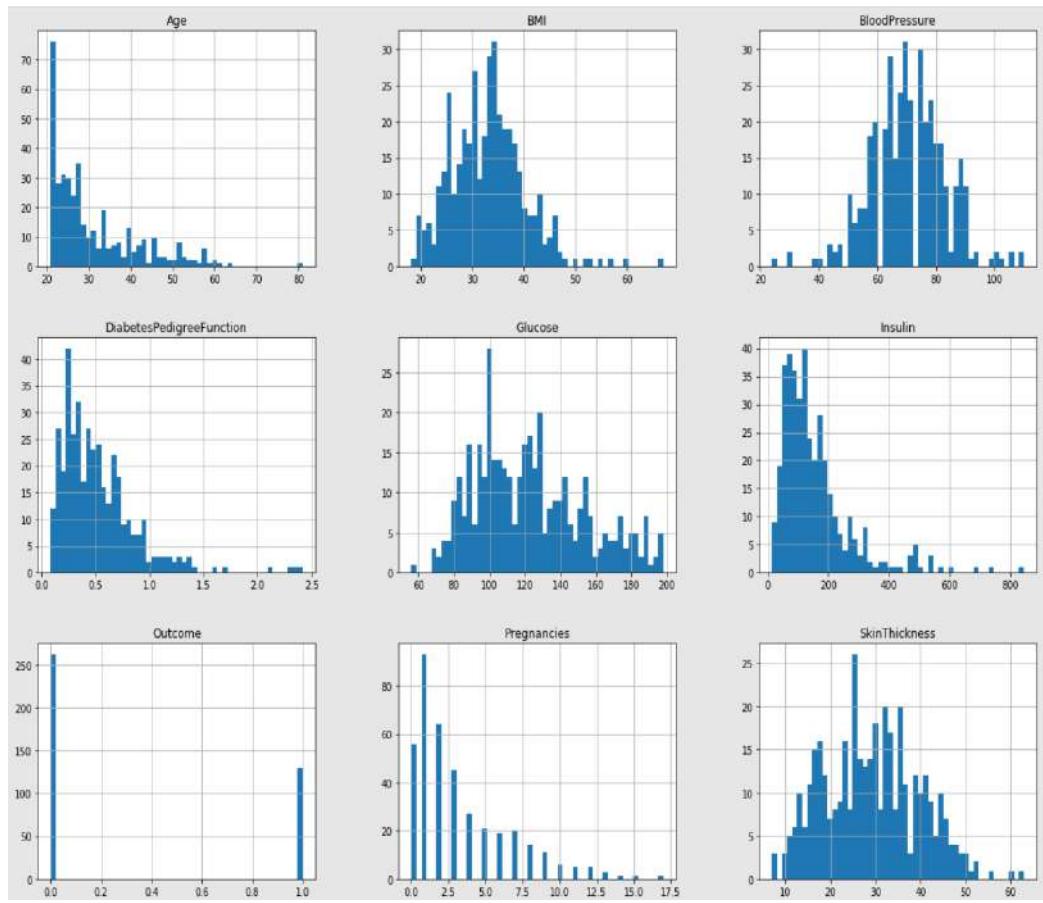


Figure.2 Input parameters of PIMA Indian dataset

3.3 Data Pre-processing

Data pre-processing is most important process. Mostly healthcare related data contains missing value and other impurities that can cause effectiveness of data. To improve quality and effectiveness obtained after mining process, Data pre-processing is done. To use Machine Learning Techniques on the dataset effectively this process is essential for accurate result and successful prediction. For Pima Indian diabetes dataset, we need to perform pre-processing in two steps.

Missing Values removal- Remove all the instances that have zero (0) as worth. Having zero as worth is not possible. Therefore, this instance is eliminated. Through eliminating irrelevant features/instances we make feature subset and this process is called features subset selection, which reduces dimensionality of data and help to work faster.

Splitting of data- After cleaning the data, data is normalized in training and testing the model. When data is spitted then we train algorithm on the training data set and keep test data set aside. This training process will produce the training model based on logic and algorithms and values of the feature in training data. Basically, aim of normalization is to bring all the attributes under same scale.

3.4 Classifiers

A classifier is a tool in machine learning that proceeds a group of data demonstrating the objects we need to classify and tries to forecast which class the new data belongs to. The classification objective set for this study is to achieve enhanced accuracy by using SVC, Decision Tree Classifier, K-Neighbours Classifier, Logistic Regression, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier and determine which one suits the most for diabetes classification technique. The classifiers we are selected to use in this study are ranked among the top ten best classifiers especially Support vector machine. These classifiers are selected on the bases of their strengths described below and also due to their frequent use in previous research studies.

3.4.1 Correlation Matrix

Correlation is an indication about the changes between two variables. We can plot correlation matrix to show which variable is having a high or low correlation in respect to another variable. We plot correlation matrix for the Pima Indian Diabetes dataset. From the above output of correlation matrix, we can see that it is symmetrical i.e., the

bottom left is same as the top right. It is also observed that each variable is positively correlated with each other. Glucose has the highest correlation with our target variable Outcome, followed by BMI. While, Blood Pressure and Skin Thickness has the lowest correlation.

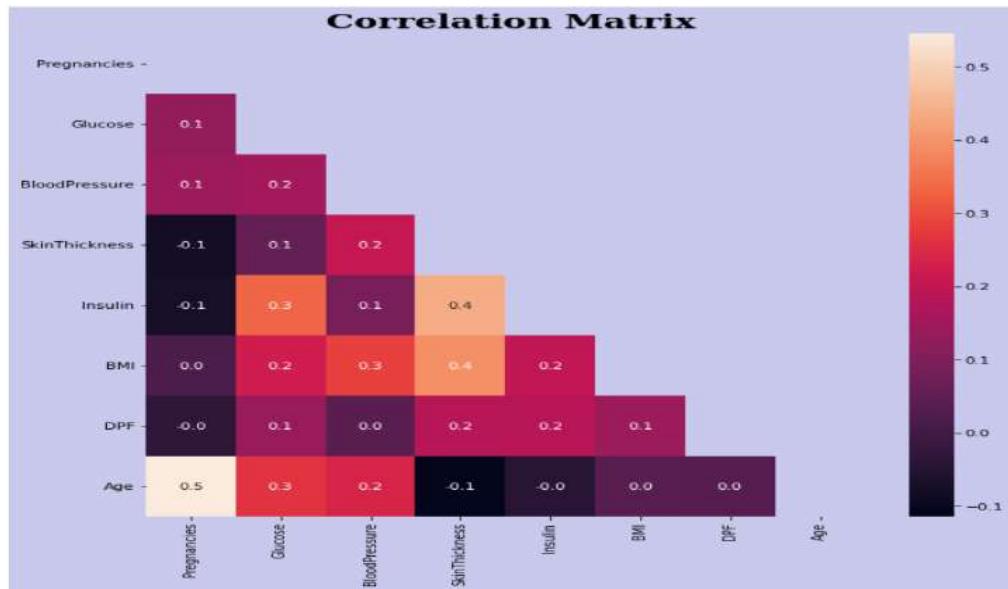


Figure 3: Correlation Matrix

3.4.2 Confusion Matrix

Summarize the result of a machine learning model by showing the correlation between the label and the model's classification. Considering the following confusion matrix that describes my train model that the model correctly classified 149 as Diabetes patient (TP) and 19 are incorrectly classified. Similarly, 47 (TN) no diabetes classified correctly where 45 incorrectly classified (FP).

Confusion Matrix Diagram

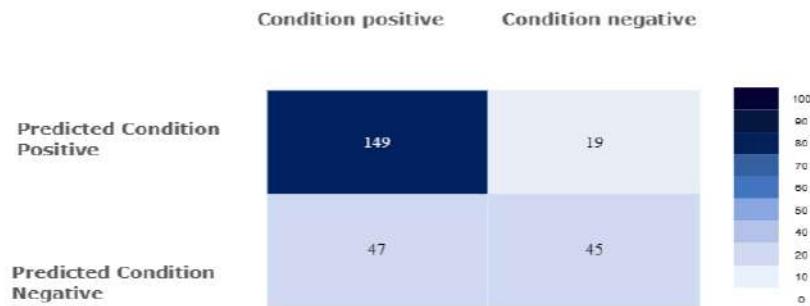


Figure 4: Confusion Matrix

Table 2. Performance metrics of various classifiers technique in the merged dataset

	PRECISION	RECALL	F1-SCORE
0	0.79	0.88	0.81
1	0.77	0.57	0.60
ACCURACY			0.79
MACRO AVG	0.76	0.70	0.71
WEIGHTED AVG	0.76	0.75	0.75

4 Results and Discussion

This section presents the results and discussion of the proposed automatic diabetes prediction system. First, the performance of various machine learning techniques is discussed. Next, the implemented website framework and Android smartphone application are demonstrated. We used precision, recall, F1 score, and classification accuracy to evaluate various ML models. We define the accuracy as mean average precision considering three concepts of measurement precision, recall and F1 score. The mathematical process of the evaluation given below,

Precision provided the total amounts of correctly predicted cases turned out to be positive. We could measure the model reliability with this precision value,

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

Recall gives the correctness of a model, total positive cases the model able to predict correctly,

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

To get a combined idea about those two values we used F1-score. It is finding the harmonic mean from precision and recall value.

$$F1 = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (3)$$

This is how I will calculate the accuracy of a model,

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+FP+TN} \quad (4)$$

where TP denotes the model is predicting positive, and the result is also positive. FP indicates the positive prediction of the model, but the result is negative. TN expresses the model is predicting negative, and the result is also negative. FN indicates the model predicts negative, but the result is positive.

Table 3 compares different performance metrics of various classifiers for the PIMA Indian dataset. According to this table, the Support vector classifier achieved the best overall performance with 79% accuracy and 0.78 and 0.79 F1 score and precision respectively.

Table 3. Performance metrics of various classifiers technique in the merged dataset

Classifier	Precision	Recall	F1 Score	Accuracy
Support vector	0.79	0.78	0.78	79%
Logistic regression	0.78	0.77	0.77	78%
Decision tree	0.74	0.73	0.73	74%
Random forest	0.77	0.76	0.76	77%
AdaBoost	0.77	0.75	0.75	77%

Gradient Boosting	0.78	0.77	0.77	78%
K-Neighbour	0.78	0.77	0.77	78%

5 CONCLUSIONS

Diabetes can be a reason for reducing life expectancy and quality. Predicting this chronic disorder earlier can reduce the risk and complications of many diseases in the long run. In this paper, an automatic diabetes prediction system using various machine learning approaches has been proposed. The open-source Pima Indian dataset of female patients have been used in this work. Pre-processing techniques have been applied to handle the issue of imbalanced class problems. These models are applied to the dataset defined in two ways: training data are kept separate from testing data. Furthermore, ten-fold cross-validation methods are applied to measure the accuracy of models. This research paper reported different performance metrics, that is, precision, recall, accuracy, F1 score, for various machine learning and ensemble techniques. The Support vector classifier achieved the best performance with 79% accuracy and an F1 score of 0.79 and 0.84, respectively. Next, the domain adaptation technique has been applied to demonstrate the versatility of the proposed prediction system. Finally, the best-performed Tikinder framework has been deployed into a website and smartphone application to predict diabetes instantly.

REFERENCES

1. Schulz LO, Bennett PH, Ravussin E, Kidd JR, Kidd KK, Esparza J, Valencia ME (2006) Effects of traditional and western environments on prevalence of type 2 diabetes in Pima Indians in Mexico and the US. *Diabetes Care* 29(8):1866–1871
2. Smith JW, Everhart JE, Dickson WC, Knowler WC, Johannes RS (1988) Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: Proceedings of the annual symposium on computer application in medical care, pp 261–265

3. Larabi-Marie-Sainte S, Aburahmah L, Almohaini R, Saba T (2019) Current techniques for diabetes prediction: review and case study. *Appl Sci* 9(21):4604
4. Pratap Singh R, Javaid M, Haleem A, Vaishya R, Ali S (2020) Internet of Medical Things (IoMT) for orthopaedic in COVID-19 pandemic: roles, challenges, and applications. *J Clin Orthop Trauma* 11(4):713–717. <https://doi.org/10.1016/j.jcot.2020.05.011>
5. Pustokhina IV, Pustokhin DA, Gupta D, Khanna A, Shankar K, Nguyen GN (2020) An effective training scheme for deep neural network in edge computing enabled Internet of Medical Things (IoMT) systems. *IEEE Access* 8:107112–107123. <https://doi.org/10.1109/ACCESS.2020.3000322>
6. Alsubaei F, Abuhussein A, Shandilya V, Shiva S (2019) IoMT-SAF: Internet of Medical Things security assessment framework. *Internet Things* 8:100123. <https://doi.org/10.1016/j.iot.2019.100123>
7. Khan SU, Islam N, Jan Z, Din IU, Khan A, Faheem Y (2019) An e-Health care services framework for the detection and classification of breast cancer in breast cytology images as an IoMT application. *FuturGenerComputSyst* 98:286–296. <https://doi.org/10.1016/j.future.2019.01.033>
8. Divya K, Sirohi A, Pande S, Malik R (2021) An IoMT assisted heart disease diagnostic system using machine learning techniques. In: Hassanien AE, Khamparia A, Gupta D, Shankar K, Slowik A (eds) *Cognitive Internet of Medical Things for smart healthcare*, vol 311. Springer, New York, pp 145–161. https://doi.org/10.1007/978-3-030-55833-8_9
9. Kumar PM, Devi Gandhi U (2018) A novel three-tier Internet of Things architecture with machine learning algorithm for early detection of heart diseases. *ComputElectrEng* 65:222–235. <https://doi.org/10.1016/j.compeleceng.2017.09.001>
10. Kaur H, Kumari V (2020) Predictive modelling and analytics for diabetes using a machine learning approach. In: *Applied computing and informatics*, ahead-of-print (ahead-of-print). <https://doi.org/10.1016/j.aci.2018.12.004>
11. Adeniyi EA, Ogundokun RO, Awotunde JB (2021) IoMT-based wearable body sensors network healthcare monitoring system. In: Marques G, Bhoi

- AK, de Albuquerque VHC, Hareesha KS (eds) IoT in healthcare and ambient assisted living, vol 933. Springer, Singapore, pp 103–121. https://doi.org/10.1007/978-981-15-9897-5_6
12. Komi M, Li J, Zhai Y, Zhang X (2017) Application of data mining methods in diabetes prediction. In: 2017 2nd international conference on image, vision and computing (ICIVC), Chengdu, China, pp 1006–1010. <https://doi.org/10.1109/ICIVC.2017.7984706>
 13. Samant P, Agarwal R (2018) Machine learning techniques for medical diagnosis of diabetes using iris images. *Comput Methods Programs Biomed* 157:121–128. <https://doi.org/10.1016/j.cmpb.2018.01.004>
 14. Quinlan JR (1996) Learning decision tree classifiers. *ACM ComputSurv (CSUR)* 28(1):71–72
 15. Saxena R (2017) How decision tree algorithm works. <https://dataaspirant.com/2017/01/30/how-decision-tree-algorithm-works/>. Accessed Apr 40.
 16. Rochmawati N, Hidayati HB, Yamasari Y, Yustanti W, Rakhmawati L, Tjahyaningtjas HPA, Anistyasari Y (2020) Covid symptom severity using decision tree. In: 2020 third international conference on vocational education and electrical engineering (ICVEE). Surabaya, Indonesia, pp 1–5. <https://doi.org/10.1109/ICVEE50212.2020.9243246>
 17. Gomathi S, Narayani V (2015) Monitoring of Lupus disease using decision tree induction classification algorithm. In: 2015 international conference on advanced computing and communication systems. Coimbatore, India, pp 1–6. <https://doi.org/10.1109/ICACCS.2015.7324054>
 18. Abdar M, Nasarian E, Zhou X, Bargshady G, Wijayaningrum VN, Hussain S (2019) Performance improvement of decision trees for diagnosis of coronary artery disease using multi filtering approach. In: 2019 IEEE 4th international conference on computer and communication systems (ICCCS). Singapore, pp 26–30. <https://doi.org/10.1109/CCOMS.2019.8821633>



Springer

<https://icoisc.org/2023/>