

**Algorithms for Intelligent Systems**

*Series Editors:* Jagdish Chand Bansal · Kusum Deep · Atulya K. Nagar

Aditya Kumar Singh Pundir  
Anupam Yadav  
Swagatam Das *Editors*

# Recent Trends in Communication and Intelligent Systems

Proceedings of ICRTCIS 2023



Springer

# **Algorithms for Intelligent Systems**

## **Series Editors**

Jagdish Chand Bansal, Department of Mathematics, South Asian University,  
New Delhi, Delhi, India

Kusum Deep, Department of Mathematics, Indian Institute of Technology Roorkee,  
Roorkee, Uttarakhand, India

Atulya K. Nagar, School of Mathematics, Computer Science and Engineering,  
Liverpool Hope University, Liverpool, UK

This book series publishes research on the analysis and development of algorithms for intelligent systems with their applications to various real world problems. It covers research related to autonomous agents, multi-agent systems, behavioral modeling, reinforcement learning, game theory, mechanism design, machine learning, meta-heuristic search, optimization, planning and scheduling, artificial neural networks, evolutionary computation, swarm intelligence and other algorithms for intelligent systems.

The book series includes recent advancements, modification and applications of the artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, fuzzy system, autonomous and multi agent systems, machine learning and other intelligent systems related areas. The material will be beneficial for the graduate students, post-graduate students as well as the researchers who want a broader view of advances in algorithms for intelligent systems. The contents will also be useful to the researchers from other fields who have no knowledge of the power of intelligent systems, e.g. the researchers in the field of bioinformatics, biochemists, mechanical and chemical engineers, economists, musicians and medical practitioners.

The series publishes monographs, edited volumes, advanced textbooks and selected proceedings.

**Indexed by zbMATH.**

**All books published in the series are submitted for consideration in Web of Science.**

Aditya Kumar Singh Pundir · Anupam Yadav ·  
Swagatam Das  
Editors

# Recent Trends in Communication and Intelligent Systems

Proceedings of ICRTCIS 2023



Springer

*Editors*

Aditya Kumar Singh Pundir  
Arya College of Engineering and IT  
Jaipur, India

Anupam Yadav  
Dr BR Ambedkar NIT Jalandhar  
Jalandhar, Punjab, India

Swagatam Das  
Electronics and Communication Sciences  
Unit  
Indian Statistical Institute  
Kolkata, India

ISSN 2524-7565                   ISSN 2524-7573 (electronic)  
Algorithms for Intelligent Systems  
ISBN 978-981-99-5791-0       ISBN 978-981-99-5792-7 (eBook)  
<https://doi.org/10.1007/978-981-99-5792-7>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2023

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.  
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721,

Paper in this product is recyclable.

# **Conference Program Committee**

## **International Advisory Committee**

Dr. Bimal K. Bose, Em. Prof., E & C Engineering, The University of Tennessee, Knoxville, USA

Dr. Subramaniam Ganesan, Prof., E & C Engineering, Oakland University, USA

Dr. L. M. Patnaik, Adjunct Prof. and INSA Senior Scientist & Former Prof., Department of Electrical Systems Engineering, IIS, Bangalore, India

Dr. Ramesh Agarwal, Prof., School of Engineering & Applied Science, Washington University, USA

Dr. Vincenzo Piuri, Prof., University of Milan, Italy

Dr. Ashoka Bhat, Prof., Department of Electrical and Computer Engineering, University of Victoria, Canada

Prof. Akhtar Kalam, Head of External Engagement, Victoria University, Victoria, Australia

Dr. M. H. Rashid, Prof., E & C Engineering, University of West Florida, USA

Dr. Fushuan Wen, Director, Zhejiang University-Insigma, Joint Research Center for Smart Grids, China

Prof. Ir. Dr. N. A. Rahim, Director, UMPEDAC, UM, Kuala Lumpur, Malaysia

Prof. Rafael F. S. Caldeirinha, Coordinator Prof., Polytechnic of Leiria and Instituto de Telecomunicações, Portugal

Dr. Tarek Bouktir, Prof., EE, University of Setif, Algeria

Dr. Pietro Savazzi, Telecommunications and Remote Sensing Laboratory, Department of Electrical, Biomedical and Computer Engineering, University of Pavia, Italy

Dr. Ouri Wolfson, President and Chief Scientist, Pirouette Software, Inc., Fellow of the ACM, IEEE, AAAS, Chicago, IL, USA

Dr. Álvaro Rocha, Prof., Information Systems at the University of Lisbon—ISEG, Portugal

Dr. Pavel Loskot, Assoc. Prof., ZJU-UIUC Institute, Haining, Zhejiang, China

Dr. S. Mekhilef, PEARL, EE, University of Malaya, Kuala Lumpur, Malaysia

Dr. Mehmet Emir Koksal, Assoc. Prof., Ondokuz Mayis University, Department of Mathematics, Atakum, Samsun, Turkey

Dr. Biplob Ray, Senior Lecturer—ICT | Discipline Lead—Postgraduate Foundation, School of Engineering and Technology, CQ, University Australia, Melbourne Campus

## National Advisory Committee

Dr. S. N. Joshi, Em. Scientist, CSIR-CEERI Pilani, India

Dr. Vineet Sahula, Prof., ECE, MNIT, Jaipur, India

Dr. K. J. Rangra, Chief Scientist and Prof. AcSIR, CSIR-CEERI, Pilani, India

Dr. R. K. Sharma, Sr. Principal Scientist, Prof., AcSIR, CSIR-CEERI, Pilani, India

Dr. Vijay Janyani, Prof., ECE, MNIT, Jaipur, India

Dr. K. R. Niazi, Prof., EE, MNIT, Jaipur, India

Dr. V. K. Jain, former Director Grade Scientist, Solid State Physics Lab., DRDO, India

Dr. Manoj Kumar Patairiya, Director, CSIR, NISCAIR, India

Dr. Sanjeev Mishra, Prof., UDME, RTU, Kota, India

Dr. Kailash N. Srivastava, VC, SUAS, Indore, India

Prof. Harpal Tiwari, Prof., EE, MNIT, Jaipur, India

Dr. S. Gurunarayanan, Prof., BITS, Pilani, India

Dr. Ghanshyam Singh, Prof., ECE, MNIT, Jaipur India

Dr. R. Kumar, GEC, Ajmer, Rajasthan, India

Prof. Satish Kumar, Sr. Principal Scientist, AMS, Prof., AcSIR, CSIR-CSIO, Chandigarh, India

Dr. Kota Srinivas, Chief Scientist, CSIR-CSIO, Chennai Centre, CSIR Madras Complex, Chennai, India

Sh. Anand Pathak, President, SSME, India

Sh. Ulkesh Desai, Vice President, SSME, India

Sh. Ashish Soni, Secretary, SME, India

Sh. R. M. Shah, Joint Secretary, SSME, India

Sh. VimalShah, Treasurer, SSME, India

Acharya (Er.) Daria Singh Yadav, Chairman, ISTE Rajasthan and Haryana Section, India

Er. Sajjan Singh Yadav, Chairman, IEI, Jaipur, India

Er. Gautam Raj Bhansali, Honorary Secretary, IEI, Jaipur, India

Dr. J. L. Sehgal, IEI, Jaipur, India

Smt. Annapurna Bhargava, IEI, Jaipur, India

Smt. Jaya Vajpai, IEI, Jaipur, India

Dr. Hemant Kumar Garg, IEI, Jaipur, India

Er. Gunjan Saxena, IEI, Jaipur, India

Er. Sudesh Roop Rai, IEI, Jaipur, India

Dr. Manish Tiwari, Manipal University Jaipur, India

Dr. Dinesh Yadav, Manipal University Jaipur, India

Dr. Jitendra Kumar Deegwal, Principal, Government Women Engineering, College, Ajmer, India

## **Organizing Committee**

### **Chief Patrons**

Smt. Madhu Malti Agarwal, Chairperson, Arya Group, India

Er. Anurag Agarwal, Chairman, ACEIT, Jaipur, India

### **Patrons**

Prof. Arun Kr. Arya, Principal, ACEIT, Jaipur, India

## **General Chair**

Dr. Swagatam Das, ISI Kolkatta, India

Dr. Harish Sharma, RTU Kota, India

Dr. Anupam Yadav, NIT Jalandhar, India

Dr. Vibhakar Pathak, ACEIT, Jaipur, India

## **Conveners**

Dr. Kirti Vyas, ACEIT, Jaipur, India

Dr. Chhavi Saxena, ACEIT, Jaipur, India

Mr. Sachin Chouhan, ACEIT, Jaipur, India

Er. Ankit Gupta, ACEIT, Jaipur, India

## **Co-Conveners**

Ms. Chanchal Sharma, ACEIT, Jaipur, India

Mr. Amit Sharma, ACEIT, Jaipur, India

## **Organizing Chair & Secretaries**

Dr. Rahul Srivastava, ACEIT, Jaipur, India

Dr. Aditya Kr. S. Pundir, ACEIT, Jaipur, India

## **Special Session Chair**

Dr. Sarabani Roy, Jadavpur University, Kolkata, India

Dr. Nirmala Sharma, RTU Kota, India

Dr. Irum Alvi, RTU, Kota, India

Dr. S. Mekhilef, University of Malaya, Malaysia

## **Local Organizing Committee Chairs**

Prof. Akhil Pandey, ACEIT, Jaipur, India

Prof. Prabhat Kumar, ACEIT, Jaipur, India

Prof. Sourabh Bhaskar, ACEIT, Jaipur, India

Prof. Shalani Bhargava, ACEIT, Jaipur, India

Shri Ramcharan Sharma, ACEIT, Jaipur, India

# Preface

This volume comprises selected and shortlisted papers presented at the 4th *International Conference on Recent Trends in Communication & Intelligent Systems (ICRT CIS 2023)*, Organized by Arya College of Engineering and Information Technology and hosted by the Department of Electronics & Communication Engineering in hybrid mode on 28th–29th April 2023. The publication partner of *ICRT CIS 2023* was *Springer India*.

The papers cover selective high-impacted areas and topics related to intelligent systems and communication networks, including *Intelligent Computing & Converging Technologies, Intelligent System Communication and Sustainable Design and Intelligent Control, Measurement & Quality Assurance*. This Algorithm for Intelligent Systems volume brings the best 19 of the presented papers. Each presents new approaches and/or evaluates methods to real-world problems and exploratory research that describes novel approaches in the field of intelligent systems.

ICRT CIS 2023 has received (all tracks) 185 submissions, 23 were shortlisted for presentation, and 19 papers were finally selected for publication in Springer proceedings. The authors and technical program committee members with wide diversity around the globe have participated from different countries, India, Nigeria, and Saudi Arabia. Nine renowned keynotes around the globe have delivered a glimpse of their current research and interacted with the participants.

We hope all the participants have enjoyed the parallel sessions and presentation tracks: two parallel tracks after the keynotes.

The editors trust this volume will be useful and interesting to readers for their research work.

Jaipur, India  
Jalandhar, India  
Kolkata, India  
June 2023

Aditya Kumar Singh Pundir  
Anupam Yadav  
Swagatam Das

# Contents

<b>Performance Analysis of UAV Routing Protocol Based on Mobility Models .....</b>	1
Kanchan Vipul Bakade and Avinash More	
<b>Evaluation of Lubricant Film Thickness in Helical Gear on Contact Line at Critical Points by Using Elastohydrodynamic Model and It's Comparison with Kisssoft Result .....</b>	15
Ashutosh Kumar and Kiran More	
<b>ABC Algorithm for Evaluating the Performance of the SVC and Optimal Power Flow .....</b>	37
N. Kalpana	
<b>ELPDI: A Novel Ensemble Learning Approach for Pulmonary Disease Identification .....</b>	49
Satwik Kulkarni, Sitanshu Hallad, Tanvi Bhujannavar, Abhishek S. Masur, Shankru Guggari, Uday Kulkarni, and S. M. Meena	
<b>JAYA-Based Task Scheduling Algorithm in Fog-Cloud Environment .....</b>	61
Nupur Jangu and Zahid Raza	
<b>BLOCK-FEMF: Efficient Forensic Evidence Management Framework Using Blockchain Technology .....</b>	71
Praveen M. Dhulavvagol, Atrey Mahadev Anagal, Vinayak Sudhakar Kone, Swaroop Anegundi, and Praveen Devadakar	
<b>Bacterial Spot Classification in Bell Pepper Plants Based on Convolutional Neural Network Architecture .....</b>	89
Midhun P. Mathew, Sudheep Elayidom, and V. P. Jagathiraj	
<b>Prediction of Bone Mineral Density Using AI to Detect Osteoporosis ...</b>	105
Ritu Chauhan, Yati Varshney, and Harleen Kaur	

<b>On the Depth of Convolutional Neural Networks for Image Classification Problems .....</b>	117
Samima Khatun, Md. Yousuf Ansari, Sk. Saidul Alam, and Ayatullah Faruk Mollah	
<b>A Modified Tilt Integral Derivative (MTID) Controller Design Using Direct Synthesis Approach .....</b>	127
Rammurti Meena, Vipin Chandra Pal, and Sudipta Chakraborty	
<b>A Novel Modified Semicircular Patch Antenna for Next Generation Communication .....</b>	139
Kiran Githala and Manisha Gupta	
<b>Panoptic Driving Perception Using Deep Learning Techniques .....</b>	149
Uday Kulkarni, Shashank Agasimani, Sagar Kabadi, P. S. Aditya, Vittal Hundarad, and Rajeshkanna Viswanathan	
<b>Schistosomiasis Prevention Undermined by Water Point Forecasting: An Approach Based on Data Fusion Model .....</b>	163
Teegwende Zougmore, Sadouanouan Malo, and Bamba Gueye	
<b>A Comparative Study of Classification Algorithms for the Prediction of Communication Effectiveness .....</b>	179
S. Suji and D. Abisha	
<b>A Dynamic Threshold-Based Cognitive Radio Energy Detection Algorithm for Multi-carrier Waveforms .....</b>	193
Jyoti Gupta, Ankit Gupta, Sachin Chauhan, and Rahul Srivastava	
<b>Comparison of Different Alphabetically Shaped Microstrip Patch Antennas in the S-Band Using HFSS .....</b>	201
Shubhi Gupta, Sachin Chauhan, Rahul Srivastava, and Ankit Gupta	
<b>Using MNIST Dataset for De-Pois Attack and Defence .....</b>	213
Chetan Niloor, Rashmi Agarwal, and Pradeepa Mishra	
<b>Direction Detection of Select Stocks with Machine Learning .....</b>	229
Anand Mohan, Rashmi Agarwal, and J. B. Simha	
<b>Prediction of Fatal Accidents in the Petrochemical Industry with Machine Learning .....</b>	245
Thiyagarajan Balasubrani, Rashmi Agarwal, and Mithun Dolthody Jayaprakash	
<b>Author Index .....</b>	263

# Performance Analysis of UAV Routing Protocol Based on Mobility Models



Kanchan Vipul Bakade and Avinash More

**Abstract** The communication between the flying birds or unmanned aerial vehicles (UAVs) a.k.a Flying Adhoc Network (FANET), is the most challenging task in an extremely vibrant environment. To overcome the problem of data dissemination in changed topology, many routing protocols are investigated and modified. The stable path and congestion avoidance are the evolving area in FANET. This paper investigates preliminary UAV routing protocol with different mobility models to examine the factors such as latency, average end-to-end (E2E) delay, packet delivery ratio (PDR), etc. in dynamic environment with numerous node count and node speed variations. The simulation results presented here show that increasing the node count from 0 to 49 keeping constant velocity of 20 m/sec reflects that throughput of Optimized Link State Routing (OLSR) is 1.7 kbps using Gauss Markov (GM) mobility model (MM), which is higher than throughput using Random Way-point (RWP) MM. For the same simulation condition, a lower average E2E delay of 0.44 ms for OLSR is achieved using GM MM. However, by increasing the node speed in the range of 20 m/sec to 100 m/sec for 50 nodes, the average throughput for Ad Hoc On-Demand Distance Vector (AODV) protocol increased to 28 kbps at 75 m/sec, which is higher than OLSR using RWP MM. The simulation achieves an end-to-end delay of 600 ms and 790 ms at 50 m/sec for AODV protocol using RWP MM and GM MM, respectively. Thus, the experiment demonstrates that GM MM with variable nodes and node speed performs better than RWP MM.

**Keywords** FANET · Mobility model · Routing protocol · UAV network

---

K. V. Bakade ( ) · A. More

Department of EXTC, SVKM's NMIMS Mukesh Patel School of Technology Management and Engineering, Mumbai 400056, India  
e-mail: [kanchan.bakade@nmims.edu](mailto:kanchan.bakade@nmims.edu)

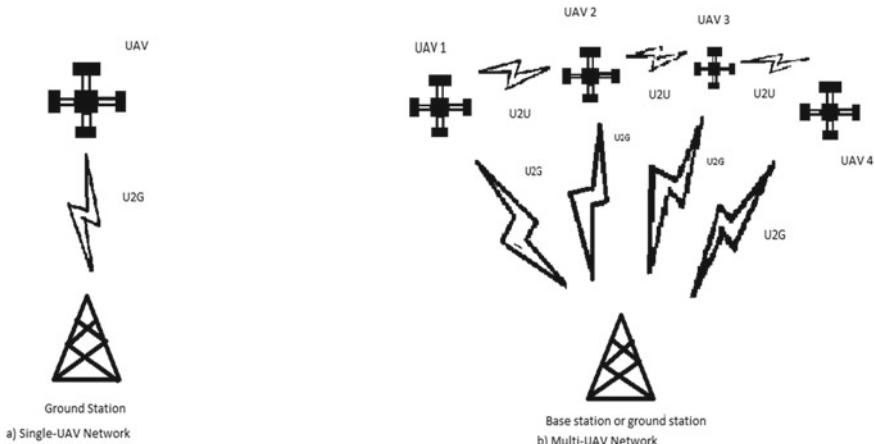
A. More  
e-mail: [avinash.more@nmims.edu](mailto:avinash.more@nmims.edu)

## 1 Introduction

Unmanned aerial vehicles (UAVs) consist of radio interfaces, microcomputers, and position locators via GPS. A group of UAVs forms a swarm to collect, forward, and share information. The swarm of UAVs can be handled or controlled automatically or remotely [1]. Due to their diverse functionality, they are used in many applications like public well-being and landslide monitoring, surveillance, disaster management, remote supervision, and monitoring using routing protocol [2–6].

The UAV networks are categorized into two types: The Single-UAV network where the UAV interacts with the terrestrial base station. They collect real-time mission information and transmit it to base station. Second, a UAV-to-UAV network where all the UAVs as well as the terrestrial base station, are linked to each other. Using multi-hop communication channels, two UAVs can interact directly with each other or indirectly with other UAVs. The link for communication in single UAV and multi-UAV networks is referred to as U2G (UAV to ground) and U2U (UAV to UAV) [1], respectively, as shown in Fig. 1.

The involvement of many UAVs will pose different issues such as a change in topology due to variable mobility [7], interchange of information, QoS, relay transmission, data loss, etc. The main concern during the operations is the node density [8], link stability, reliability, route tracing and optimization, multipath factors, propagation environment [9, 10], and dynamic behavior of the network. Hence, the incremental planning of routing protocols [11] has proven to be more important to focus on the different qualities of FANETs [12], which can prevent packet loss and adjust to variations of settings and requirements [13] in a supportive way. The routing algorithm for UAV networks is expected to attain the following properties: high throughput, minimum delay, less packet loss, minimal congestion, and low routing overhead.



**Fig. 1** UAV networks

The Protocol simulation is influenced by two factors; one is the mobility model, and another is the simulation environment. To create a FANET, each UAV must have networking components installed. A FANET's implementation is limited by the distance over which UAVs may safely communicate with one another. For UAV swarm applications, dynamic reconfiguration of routing [14–16] results in packet loss. For applications requiring precise data transmission [17, 18] between UAVs is essential, and establishing a viable FANET is difficult [19].

This paper contributes to the protocol's behaviors using the Random Waypoint mobility model (RWP MM) and Gauss Markov mobility model (GM MM) using NS3 simulator.

The paper is arranged in the following manner: Sect. 2 highlights the mobility models and routing protocols. The methodology and experimental evaluation are shown in Sect. 3. The detailed findings are discussed in Sect. 4 and concluded with observation in Sect. 5 with future directions.

## 2 Mobility Model and Routing Protocol

The maneuver of UAVs in a certain area is depicted by the FANET mobility model [20], which was customized to meet the needs of each application, resulting in better performance and flexibility. Before actual deployment and evaluation, mobility models are capable to simulate UAV behaviors in a practical approach, to produce the most accurate possible findings [21]. These mobility models are categorized as random-based mobility model, time-based mobility model, path-based mobility model, and topology-based mobility model.

- Random-Based mobility model

Each UAV chooses its actions fully independently from other UAVs at random, and the motions are also random in respect of their direction, range, and duration. This mobility model is categorized as a random direction, random waypoint, random walk, etc.

- Time-Based mobility model

The UAVs are controlled using mathematical calculations depending on the present location, previous directions, and speeds in the time-based mobility model. To guarantee that motions are revised regularly and to evade unexpected and quick variations in speed and direction, all of these factors are taken into consideration. Boundless simulation area (BSA), Gauss-Markov, Enhanced Gauss-Markov (E-GM), etc. are some time-based mobility models.

- Path-Based mobility model

In the path-based mobility model, each UAV is programmed using a predetermined trajectory to prevent random movement. The UAV can arbitrarily shift direction

and continue further after finishing this predetermined path. Semi-random circular movement, paparazzi, etc. are some mobility models that fall into this category.

- Topology-Based mobility model

The real-time control of UAV mobility is required when an application must be satisfied continuously across time. The UAVs must constantly be monitored to prevent unnecessary random motions, and they must coordinate their location with one another. Distributed pheromone repel (DPR), hybrid-Markov mobility model with pheromones (H3MP), and self-deployable point coverage (SDPC) are a few mobility models under this category.

These mobility models are applied in different routing protocols for estimating network performance and behavior [22–25]. The creation of innovative routing protocols includes UAVs with extremely dynamic topologies, optimal power usage, link failure recovery, robustness, and security. Every protocol that has been developed aims to increase the packet delivery rate while also delivering minimal delays and packet losses. The common routing protocols are Optimized Link State Routing Protocol (OLSR) [26], Destination-Sequenced Distance Vector (DSDV), Dynamic Source Routing (DSR) [27, 28], and Ad Hoc On-Demand Distance Vector (AODV) [29].

The focus of this research is on proactive and reactive topology-based routing systems, which use nodes' IP addresses to distinguish among different nodes and connections to transmit packets along the path.

### 3 Methodology and Experimental Evaluation

The Routing protocol using mobility models is used to establish the path in a dynamic environment between UAVs and ground stations either by node selection criteria in the forwarding area or with communication or sensing range. Here, the node UAVs broadcast the hello message to the nearest UAVs, which act as relay node and forward the data to the final destination UAV via a single route or multiple routes. The selection of neighbor nodes is done by examining the basic parameters like UAV distance, UAV energy, UAV storage, etc. Thus, here the developed scenarios show various node densities, link modifications, and mobility adjustments.

The implementation of the UAV routing algorithm is measured by different performance indicators, including PDR, Packet Loss Ratio (PLR), throughput, and E2E Delay. To evaluate the outcome of the prevailing protocols, Network Simulator (NS) version 3-NS-3.29 is used to simulate the UAV network [30]. Here, the developed scenarios show various node densities, link modifications, and mobility adjustments. The simulation is performed using RWP MM and GM MM. The dimensions of the simulation area utilized here are made up of length, width, and height (1000 m \*1500 m \* 1500 m). Table 1 characterizes the parameter details owned in the simulation.

**Table 1** Simulation parameters

Network simulator	NS-3.29
Routing protocols	OLSR, AODV, DSDV, DSR
Node count	50
Node speed (m/sec)	20, 50, 75, 100
MAC protocol	IEEE 802.11 b
Path loss model	Free space
Mobility model	RWP and GM
Data rate (Mbps)	11
Packet size (bytes)	2048
Pause time (sec)	0.1
Simulation time (sec)	200

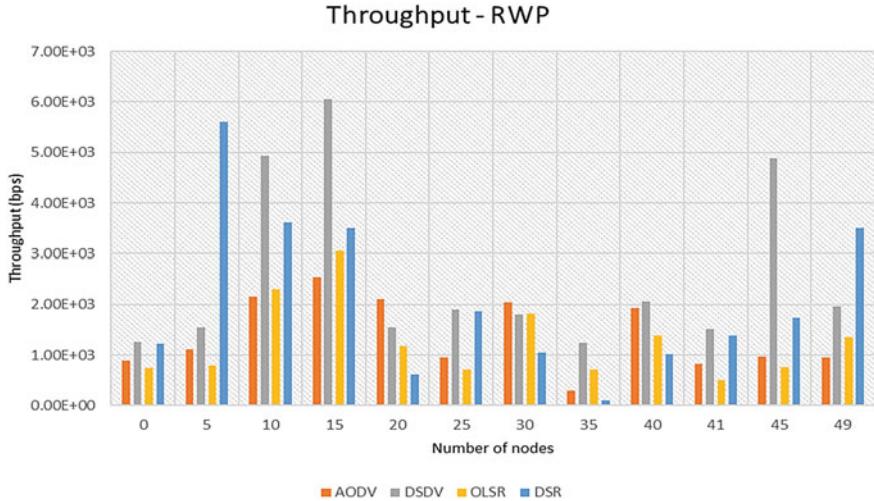
## 4 Results and Discussions

The performance of OLSR, AODV, DSDV, and DSR is analyzed and compared with the previous results by other researchers [15, 16]. The two environments were simulated, by raising the node count and the node speed.

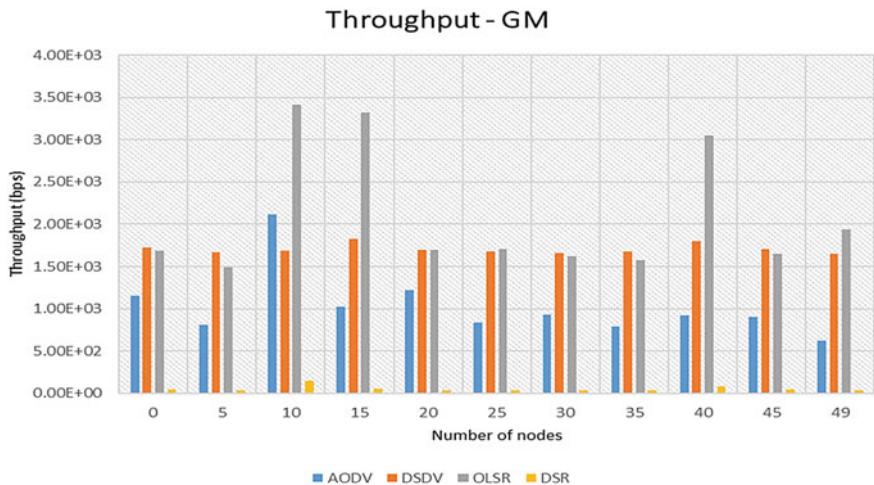
### 1. By increasing the node count

Figures 2 and 3 show the performance analysis of throughput for RWP and GM mobility models using OLSR, AODV, DSDV, and DSR protocols. In Fig. 2, all the protocols show the effective result at node 15, after that the throughput is decreasing. Suddenly DSDV and DSR show remarkable improvement for node 45 and node 49. As compared to AODV protocol [16] at node 50 the throughput is 2.55 kbps but here AODV shows a throughput of 1.5 kbps and DSR reflects 3.5 kbps using the RWP mobility model due to a change in node placement. In Fig. 3, the throughput is higher for OLSR at 1.7 kbps and lower for AODV at 0.6 kbps using the GM mobility model, which demonstrates how the nodes are more densely packed towards the middle of the simulated area and lead to difficulty in establishing the effective link.

The average E2E delay is analyzed using the UDP streams as a result of trace metrics ver1.3. The results for 50 streams are shown in Figs. 4 and 5. The data stream statistics show the stream-wise statistics of all the flows involved in the 50 nodes' example network scenario. At stream 50 in Fig. 4, AODV has the highest E2E delay of 50 ms as compared to DSDV, OLSR, and DSR, which are 24 ms, 28 ms, 20 ms using RWP model. Also, at streams 10 and 20, the delay is very low of 4 ms only. While observing stream 50 of the GM mobility model, the E2E delay of AODV is 0.8 ms, which is higher than DSDV and OLSR of 0.24 ms and 0.44 ms. Hence, GM mobility model helps in reducing the delay between the streams and is significant for link analysis of UAV networks.



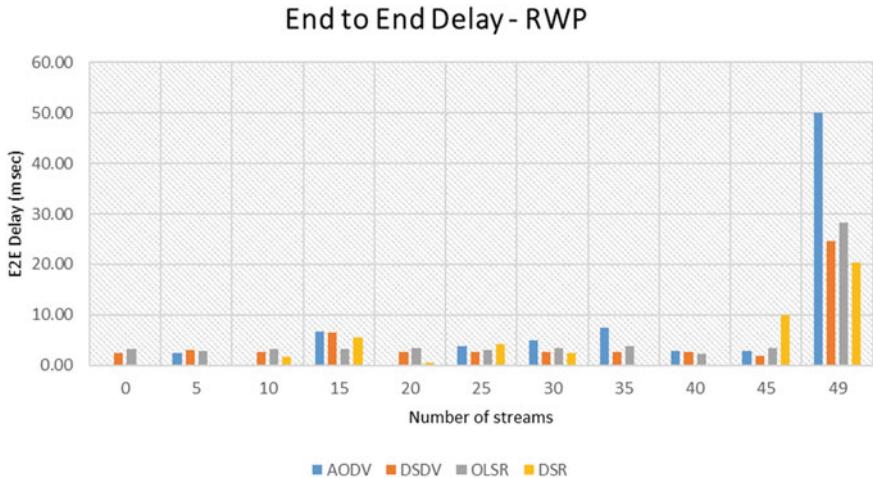
**Fig. 2** Throughput using the RWP mobility model



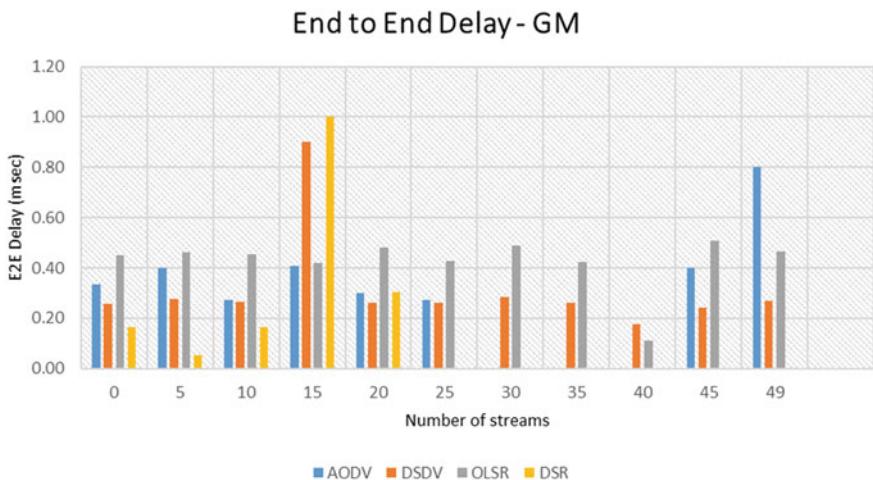
**Fig. 3** Throughput using the GM mobility model

## 2. By increasing the node speed

The PDR, average throughput, and E2E delay for both the mobility model using flow monitor flow-id (which is the total number of received bytes for each flow) are determined by increasing the node speed. This module automatically recognizes all network flows and records the parameters for later analysis, including bitrates, throughput, delays, packet sizes, packet delivery ratio, etc.

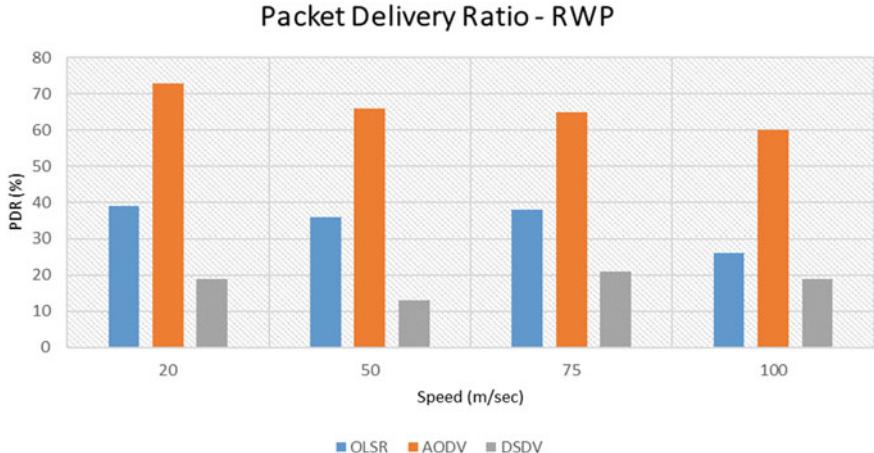


**Fig. 4** E2E delay using the RWP mobility model

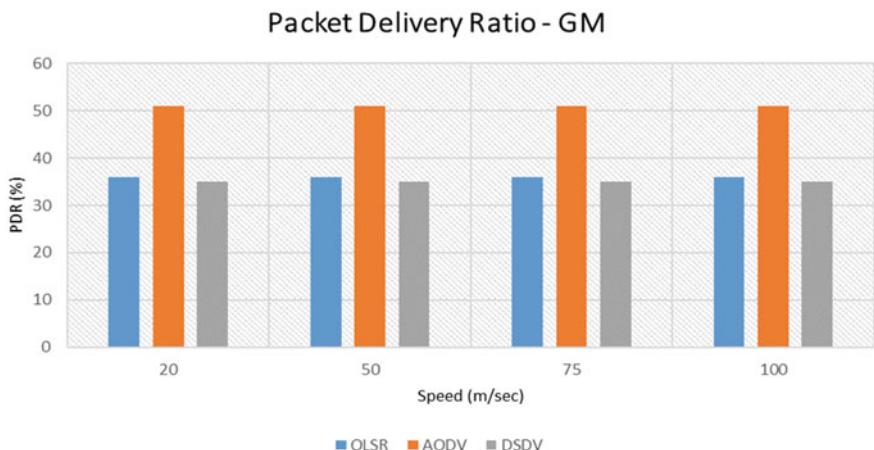


**Fig. 5** E2E delay using the GM mobility model

The node speed is varied in the range of 20–100 m/sec for OLSR, AODV, and DSDV protocols by considering 50 nodes. The PDR is decreasing with an increase in node speed using the RWP mobility model whereas it remains constant for the GM mobility model for respective protocols as shown in Figs. 6 and 7, which was lower as compared to Huang et al. [16] due to change in transmission count of link establishment, hence some improvement is required in existing algorithm to achieve greater than 80% PDR (AODV) using RWP model.

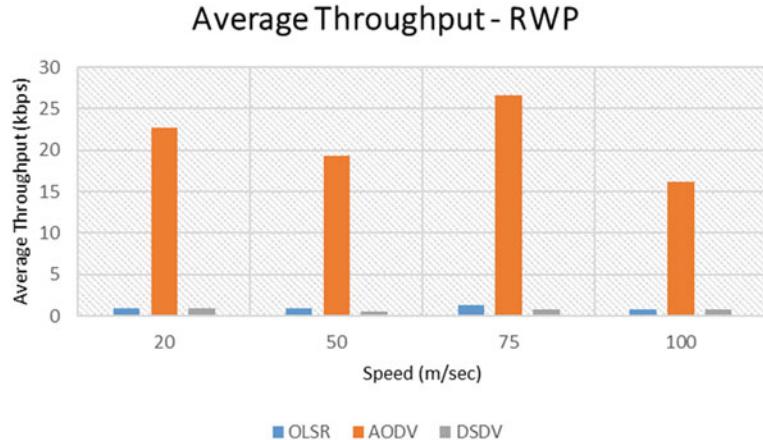


**Fig. 6** PDR using the RWP mobility model with varying speed

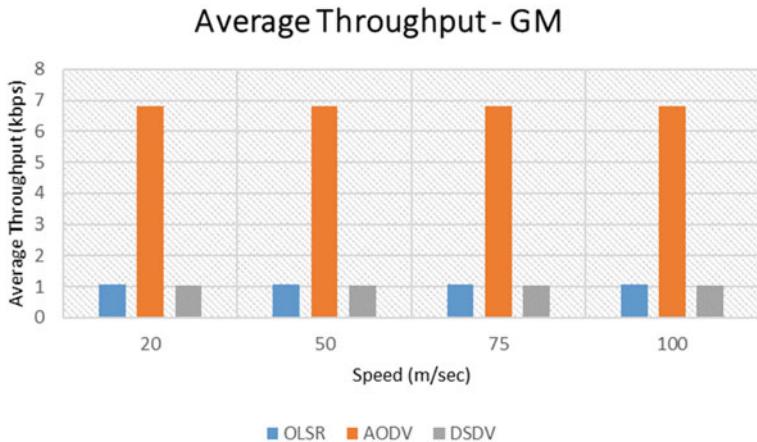


**Fig. 7** PDR using the GM mobility model with varying speed

Figures 8 and 9 show the average throughput using RWP and GM mobility model. In Fig. 8, it has been found that the average throughput increased to 28 kbps at 75 m/sec w.r.t 19 kbps at 50 m/sec for AODV protocol as compared to Huang et al. [16], which gave 2.6 kbps throughput at 40 m/sec. Also, the OLSR and DSDV results are not satisfactory due to the overflow of data and failure in link connectivity. Hence, these protocols need to be modified to attain acceptable results. Whereas the achievable average throughput was constant at variable speeds as 1 kbps, 6.9 kbps, and 1 kbps for OLSR, AODV, and DSDV protocols using the GM mobility model due to constant link connectivity as shown in Fig. 9.

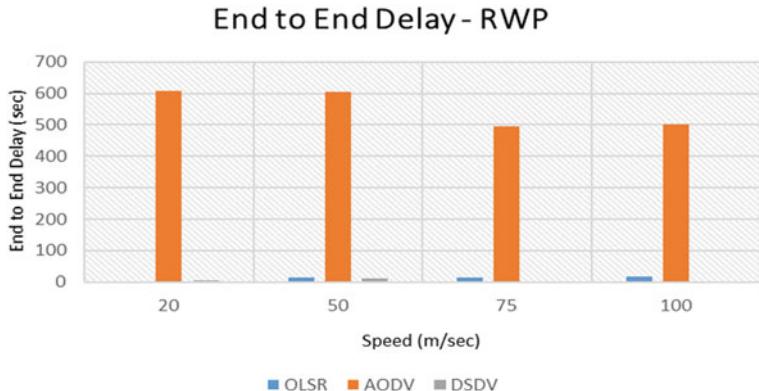


**Fig. 8** Average throughput using the RWP mobility model with varying speed

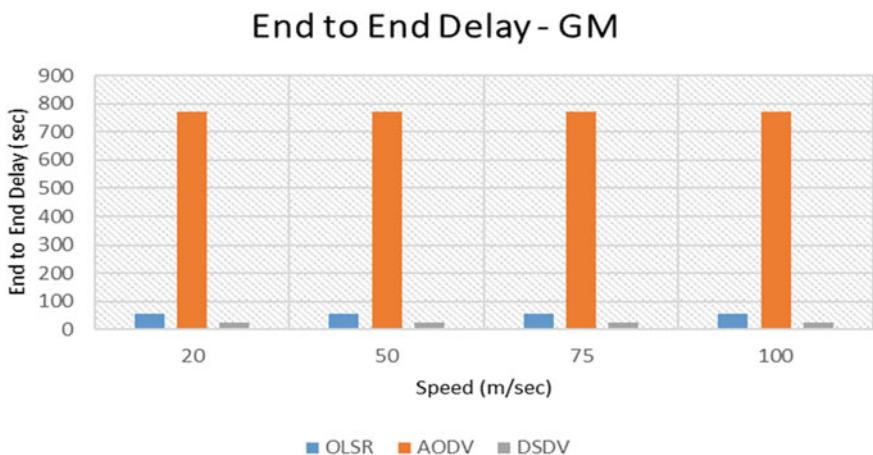


**Fig. 9** Average throughput using the GM mobility model with varying speed

The E2E delay for variable speed is shown in Figs. 10 and 11. The OLSR and DSDV protocol shows a significant drop in comparison to the AODV protocol for variable speed due to its predetermined link information. The achieved E2E delay of the AODV protocol is 600 ms and 790 ms at 20 m/sec using RWP and GM mobility model. Thus, AODV protocol must be enhanced to shrink the delay in end-to-end data transfer and to prevent data loss during changes in link connectivity, thus reducing the network overhead.



**Fig. 10** End-to-end delay using the RWP mobility model with varying speed



**Fig. 11** End-to-end delay using the GM mobility model with varying speed

The overall observation of referred protocol of Rahamani et al. [15], and Huang et al. [16] with the simulated results of OLSR and AODV protocol using RWP and GM mobility model are tabulated in Tables 2 and 3. Due to variations in node placement and link connectivity in FANET environments, the results can vary and be closer to an approximate value.

**Table 2** Comparison table of 50 nodes

Protocol	Throughput (kbps)	Delay (msec)
OLSR-RWP [15]	0.055	38
OLSR-GM [15]	0.074	40
AODV-RWP [16]	2.55	8
Simulated results		
OLSR-RWP	1.5	28
OLSR-GM	1.99	0.42
AODV-RWP	1	50
AODV-GM	0.6	0.8

**Table 3** Comparison table of 50 nodes at variable speed

Protocol	Speed—20 m/sec			Speed—50 m/sec		
	PDR (%)	Average through-put (kbps)	Average E2E delay (ms)	PDR (%)	Average through-put (kbps)	Average E2E delay (ms)
AODV-RWP [16]	90	2.66	8	87	2.64	7
Simulated results						
OLSR-RWP	39	1	—	37	1	—
OLSR-GM	36	1	—	36	1	—
AODV-RWP	72	23	600	68	19	600
AODV-GM	50	7	790	50	7	790

## 5 Conclusion and Future Enhancements

Using the RWP MM and the GM MM, this study compares the conventional procedures with some improvements from [15, 16]. The different measuring parameters, like throughput, PDR, and E2E delay, were shown for an increased node count and node speed. To achieve greater stability and to enhance the performance of link connectivity, the results are quite satisfactory for the AODV, DSDV, and OLSR protocols but not for the DSR protocol.

According to the simulation results, the GM mobility model will become more stable when the node counts and node speed rise. But still, it is preferable that the network density should not be increased tremendously.

In the future, the existing protocols to be enhanced by incorporating different environmental factors suitable for UAV swarm communication, thereby increasing network stability and reliability and giving us more information on available routing paths for data transmission with minimum overhead and broadcast messages.

**Acknowledgements** The authors of this paper are thankful to L&T Mumbai, India, for their support in this presented research work.

## References

1. Arafat MY, Moh S (2019) Routing protocols for unmanned aerial vehicle networks: a survey. *IEEE Access* 7:99694–99720
2. Lansky J, Ali S, Rahmani AM, Yousefpoor MS, Yousefpoor E, Khan F, Hosseinzadeh M (2022) Reinforcement learning-based routing protocols in flying ad hoc networks (FANET): a review. *Mathematics* 10(16):3017
3. Lansky J, Rahmani AM, Malik MH, Yousefpoor E, Yousefpoor MS, Khan MU, Hosseinzadeh M (2023) An energy-aware routing method using firefly algorithm for flying ad hoc networks. *Sci Rep* 13(1):1323
4. Zhang Y, Qiu H (2023) Delay-aware and link-quality-aware geographical routing protocol for UANET via dueling deep Q-network. *Sensors* 23(6):3024
5. Horyna J, Baca T, Walter V, Albani D, Hert D, Ferrante E, Saska M (2023) Decentralized swarms of unmanned aerial vehicles for search and rescue operations without explicit communication. *Auton Robot* 47(1):77–93
6. Kumar S, Raw RS, Bansal A, Mohammed MA, Khuwuthyakorn P, Thinnukool O (2021) 3D location oriented routing in flying ad-hoc networks for information dissemination. *IEEE Access* 9:137083–137098
7. Nawaz H, Ali HM, Laghari AA (2021) UAV communication networks issues: a review. *Arch Comput Methods Eng* 28(3):1349–1369
8. Oubbat OS, Atiquzzaman M, Lorenz P, Tareque MH, Hossain MS (2019) Routing in flying ad hoc networks: survey, constraints, and future challenge perspectives. *IEEE Access* 7:81057–81105
9. Chen Y, Zhao N, Ding Z, Alouini MS (2018) Multiple UAVs as relays: multi-hop single link versus multiple dual-hop links. *IEEE Trans Wireless Commun* 17(9):6348–6359
10. Bae JH, Kim YS, Hur N, Kim HM (2018) Study on air-to-ground multipath channel and mobility influences in UAV based broadcasting. In: 2018 International conference on information and communication technology convergence (ICTC). IEEE, pp 1534–1538
11. Khan S, Khan MZ, Khan P, Mehmood G, Khan A, Fayaz M (2022) An ant-hocnet routing protocol based on optimized fuzzy logic for swarm of UAVs in FANET. *Wireless Commun Mobile Comput*
12. Cumino P, Lobato Junior W, Tavares T, Santos H, Rosário D, Cerqueira E, Gerla M et al (2018) Cooperative UAV scheme for enhancing video transmission and global network energy efficiency. *Sensors* 18(12):4155
13. Srivastava A, Prakash J (2021) Future FANET with application and enabling techniques: atomization and sustainability issues. *Comput Sci Rev* 39:100359
14. Tuli EA, Golam M, Kim DS, Lee JM (2022) Performance enhancement of optimized link state routing protocol by parameter configuration for UANET. *Drones* 6(1):22
15. Rahmani AM, Ali S, Yousefpoor E, Yousefpoor MS, Javaheri D, Lalbakhsh P, Lee SW (2022) OLSR+: a new routing method based on fuzzy logic in flying ad-hoc networks (FANETs). *Veh Commun*:100489
16. Huang J, Zan F, Liu X, Chen D (2022) Wireless communications and mobile computing UAV routing protocol based on link stability and selectivity of neighbor nodes in ETX metrics. *Wireless Commun Mobile Comput*
17. Garg S, Ihler A, Bentley ES, Kumar S (2022) A cross-layer, mobility and congestion-aware routing protocol for UAV networks. *IEEE Trans Aerosp Electr Syst*

18. Kim T, Lee S, Kim KH, Jo YI (2023) FANET routing protocol analysis for Multi-UAV-based reconnaissance mobility models. *Drones* 7(3):161
19. Bekmezci I, Sahingoz OK, Temel S (2013) Flying ad-hoc networks (FANETs): a survey. *Ad Hoc Netw* 11(3):1254–1270
20. Zafar W, Khan BM (2016) Flying ad-hoc networks: Technological and social implications. *IEEE Technol Soc Mag* 35(2):67–74
21. Dentler J, Rosalie M, Danoy G, Bouvry P, Kannan S, Olivares-Mendez MA, Voos H (2019) Collision avoidance effects on the mobility of a UAV swarm using chaotic ant colony with model predictive control. *J Intell Rob Syst* 93(1):227–243
22. Gangopadhyay S, Jain VK (2023) A position-based modified OLSR routing protocol for flying ad hoc networks. *IEEE Trans Veh Technol*
23. Paredes WD, Kaushal H, Vakilinia I, Prodanoff Z (2023) LoRa technology in flying ad hoc networks: a survey of challenges and open issues. *Sensors* 23(5):2403
24. Chriki A, Touati H, Snoussi H, Kamoun F (2019) FANET: communication, mobility models and security issues. *Comput Netw* 163:106877
25. Singh R, Sukapuram R, Chakraborty S (2023) A survey of mobility-aware multi-access edge computing: challenges, use cases and future directions. *Ad Hoc Netw* 140:103044
26. Chen X, Tian S, Nguyen K, Sekiya H (2021) Decentralizing private blockchain-IoT network with OLSR. *Future Internet* 13(7):168
27. Varshney T, Katiyar A, Sharma P (2014) Performance improvement of MANET under DSR protocol using swarm optimization. In: 2014 International conference on issues and challenges in intelligent computing techniques (ICICT). IEEE, pp 58–63
28. Khare VR, Wang FZ, Wu S, Deng Y, Thompson C (2008) Ad-hoc network of unmanned aerial vehicle swarms for search & destroy tasks. In: 2008 4th International IEEE conference intelligent systems (vol 1). IEEE, pp 6–65
29. Moudni H, Er-rouidi M, Mouncef H, El Hadadi B (2016) Performance analysis of AODV routing protocol in MANET under the influence of routing attacks. In: 2016 International conference on electrical and information technologies (ICEIT). IEEE, pp 536–542
30. Riley GF, Henderson TR (2010) The ns-3 network simulator. In: Modeling and tools for network simulation. Springer, Berlin, Heidelberg, pp 15–34

# Evaluation of Lubricant Film Thickness in Helical Gear on Contact Line at Critical Points by Using Elastohydrodynamic Model and It's Comparison with KISSsoft Result



Ashutosh Kumar and Kiran More 

**Abstract** Helical gears in pairs develop an elliptical shape in contact areas when they are in mesh. Major and Minor axis of these elliptical contact are large and sufficient to allow the entrance of the lubricant. This usually goes through the minor axis of ellipse. Minor axis represents the direction of rolling and sliding. The critical path of the gear mesh is considered as the contact line along with various activities that take place. It is important to find the lubricant film thickness (FT) on the contact line at critical points starting from the start of the active profile to the end of the active profile. This distributed the complete meshing line in five points which is the area of interest for this research. Elastohydrodynamic Lubrication (EHL) model is developed first and then it is solved by using Mathcad Prime software to find out the critical points oil FT. Oil FT at all the five points on the contact line is determined. To perform this a sample case is considered. There are many research studies available for the Elastohydrodynamis lubrication model, however the availability of research specially with the points where the stress level changes is very rare. This research has followed the analytical way to identify the critical points and then established the Elastohydrodynamic model for these points to find the variation of lubrication oil thickness. In order to ensure the accuracy of the EHL model KISSsoft software is used to calculate the oil FT at the above said five points and the results are compared.

**Keywords** EHL · Contact line · KISSsoft

---

A. Kumar · K. More ()

Department of Mechanical Engineering, School of Engineering & Technology, D. Y. Patil University, Ambi, Pune, Maharashtra 410506, India

e-mail: [Kiran.imagine67@gmail.com](mailto:Kiran.imagine67@gmail.com)

## 1 Introduction

The contact pattern for spur gear meshing in pair forms contact over the entire face width along with the sudden ending of meshing at the end of cycle. This condition of meshing is different in case of the helical gear pair where the contact starts as the point of contact and it gradually spans over the face width. This contact does not end abruptly but it gets subsequently reduced over a contact line. This cycle gets finished as a contact point on mating the tooth face. The contact line in case of spur gear is parallel to the axis of the shaft whereas it is diagonal for the helical gear. This for a helical gear pair where geometric and kinematic conditions vary along the contact line.

To perform the elastohydrodynamic lubrication (EHL) model analysis of gear contacting surfaces, geometric analysis of the meshing cycle is required. However, the analysis of gear contact pair is difficult. It is because of continuous change in meshing geometry. Gears being an involute profile posses different radius of curvatures at each profile point. Hence before proceeding the gear geometry determination, it is important to review the gear contact.

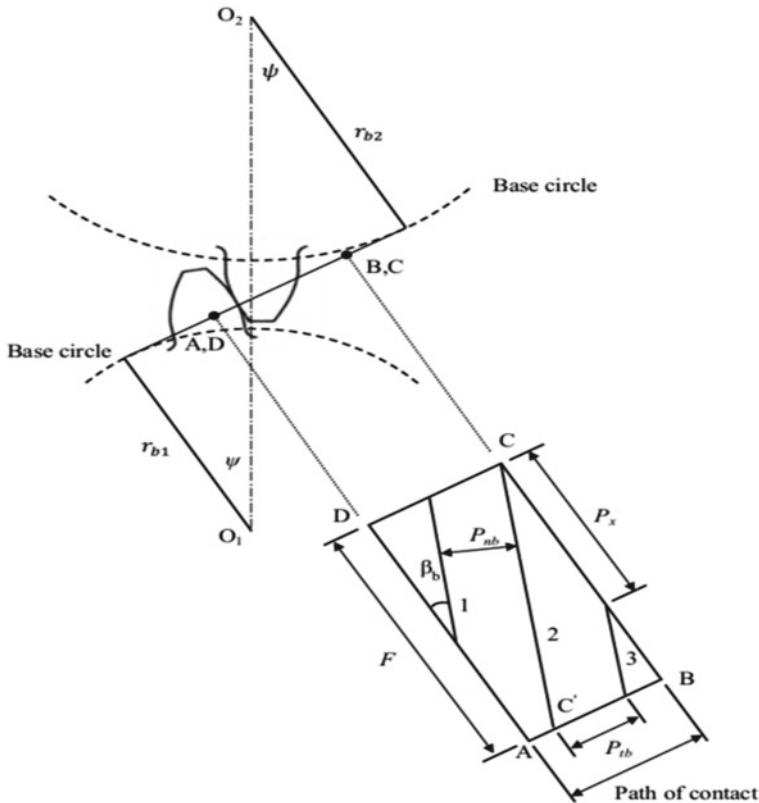
### 1.1 Contact Zone

Figure 1 shows the contact area for a helical gear pair. The rectangular section shown is the contact zone which is tangential to the base cylinder of the gear pair, and this is also known as the contact plane. AB and CD show the limits of the end faces of the contact plane and the tip cylinder of the mating gears at DA and CB. Contact of the gear pair starts at point D and ends at point C. Path of contact zone shows the contact line 1, 2 and 3, having inclination angle as the base helix angle  $\beta_b$  to the gear axis.

From the figure it is very clear that the tooth contacts occur on a parallel straight line at a normal spacing of  $P_{nb}$ . Count of the contact lines is dependent on the transverse contact ratio value.

### 1.2 Determination of Important Point of Contact Line

Figure 2 shows the various dimensions and point on the contact line in transverse direction for the external gear pair. Distances  $C_J$  are measured from the interference point of the pinion along the line of action. Distance  $C_A$  locates the pinion start of active profile (SAP) and distance  $C_E$  locates the pinion end of active profile (EAP). The lowest and highest point of the single tooth pair contact (LPSTC and HPSTC) are located by distances  $C_B$  and  $C_D$ , respectively. The operating pitch point is shown as  $y$  with distance  $C_C$  whereas  $C_F$  shows the distance between the base circles along the line of action. By using the Fig. 2 various dimensions can be derived as below:



**Fig. 1** Contact zone in helical gear pair

$$u = \frac{z_2}{z_1} \quad (1)$$

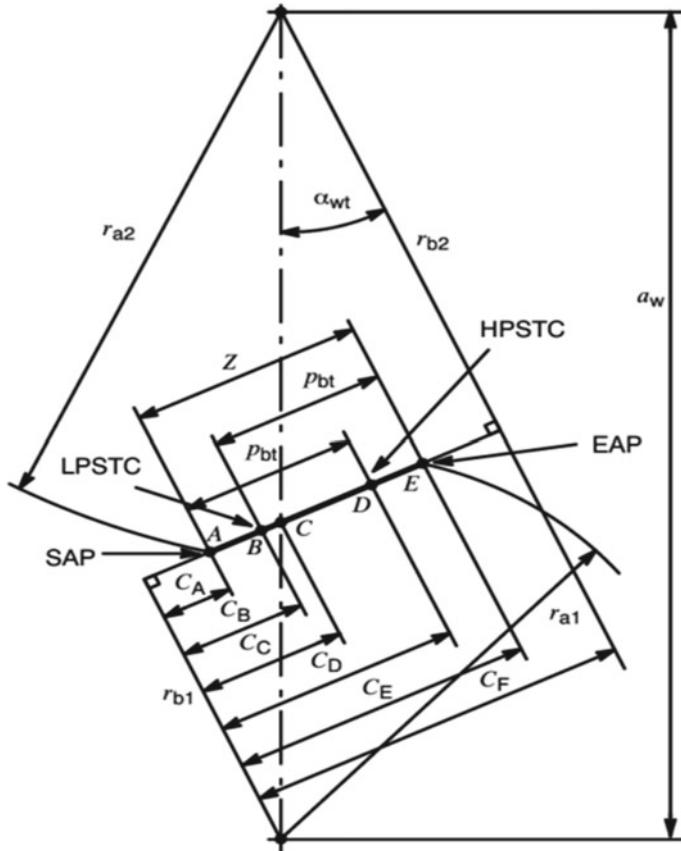
$$C_F = a_w \sin(\alpha_{wt}) \quad (2)$$

$$C_A = C_F - \sqrt{(r_{a2}^2 - r_{b2}^2)} \quad (3)$$

$$C_C = \frac{C_F}{u + 1} \quad (4)$$

$$C_D = C_A + p_{bt} \quad (5)$$

$$C_E = \sqrt{(r_{a1}^2 - r_{b1}^2)} \quad (6)$$



**Fig. 2** Distances along the line of action for external gears

$$p_{bt} = \frac{2\pi r_{b1}}{z_1} \quad (7)$$

$$C_B = C_E - p_{bt} \quad (8)$$

$$Z = C_E - C_A \quad (9)$$

After calculating the various points on the line of contact we will need to calculate the roll angles with respect to the upper calculated points on the line of contact. This can be calculated as:

$$\xi_j = \frac{C_j}{r_{b1}} \quad (10)$$

### 1.3 Contact Geometry

Ideally in the case of line contact, like contact between two cylindrical rollers, the shape of the loaded contact zone is rectangular in shape. Same conditions are not applicable for the case of helical gears as the contact zone gets deformed due to various tooth profile modifications and varying relative radius of curvature. This can create discontinuity in the tooth profile due to the local stresses. These tooth profile modifications are not considered as a part of this research work.

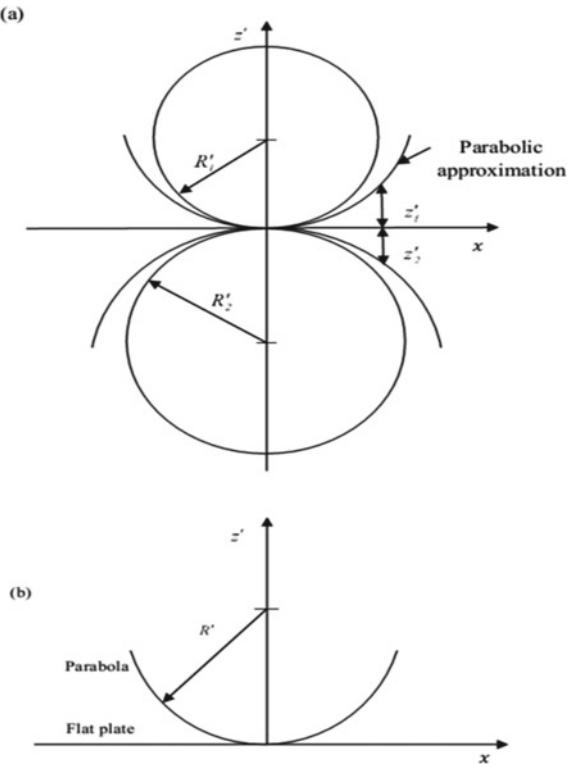
The gap between the contacting surfaces between the helical gears can be approximately represented by the plane and parabola method. This concept has good agreement in the close vicinity of the contacts as shown in Fig. 3.

Where J = A, B, C, D, E.

The gap between the contact plane and the surface as shown in Fig. 3a is determined as:

$$z'_1 = \frac{x^2}{2R'_1} \quad (11)$$

**Fig. 3** Distances along the line of action for external gears



$$z'_2 = \frac{x^2}{2R'_2} \quad (12)$$

Net gap between both the surfaces,  $h_u$ , is given as:

$$\begin{aligned} h_u &= z'_1 + z'_2 \\ h_u &= \frac{x^2}{2R'_1} + \frac{x^2}{2R'_2} = \frac{x^2}{2R'} \end{aligned} \quad (13)$$

In the above equation  $R'$  refers to the equivalent radius of curvature for both the surfaces as shown in Fig. 3b. It is shown as:

$$\frac{1}{R'} = \frac{1}{R'_1} + \frac{1}{R'_2}$$

Unlike spur gears, in helical gear pairs each point on the contact line has different radius of curvature. To approximate the helical gear pair contact, we can use approximation of the spur gear contact. For the case of spur gears the contact between them may be approximated easily by using two rollers as shown in Fig. 4.

Rollers are selected in such a way that they are equal to the tangential distance between the contact point and the base circle known as local involute radii of curvature. This dimension can be determined as:

$$R'_1 = r_{b1} \tan \alpha + s \quad (14)$$

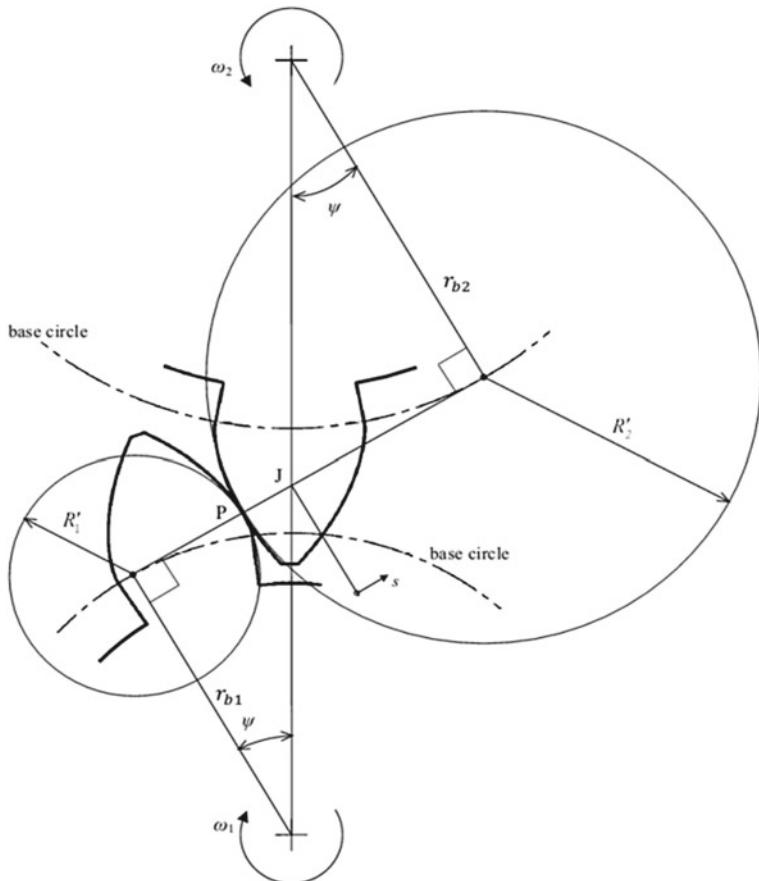
$$R'_2 = r_{b2} \tan \alpha - s \quad (15)$$

Net gap can then be calculated by putting the radii value in Eqs. (14) and (15) respectively.

Basic equations required to describe the EHL model are elastic deformation equation and hydrodynamic equation. In this section we will be discussing about the point contact problem with related equations and discuss the discretization of these equations using the Finite difference method.

## 1.4 Literature Review

Various research studies were performed globally to develop the EHL model of the involute gears. A good count of paper could be found for the case of spur gears whereas similar content for helical gears are not available. This research study is done to understand the basics and based on which the EHL model usability for the present research can be considered.



**Fig. 4** Distances along the line of action for external gears

Glovnea [1], presented a paper on A Review on experimental research carried out in past into the no-steady state Elastohydrodynamic(EHD) lubrication which included transient loading, variation in speed and variation in micro geometry. Various methods with their own advantages and disadvantages were discussed for the EHD FT measurement. Details about the behaviour of the EHD film is also discussed in transient condition. Surface related features like dent, surface finish are discussed with their effect on fluid FT. The important findings from this paper are related to the FT and its behaviour when subjected to impact loading. It also discussed about the collapse of the EHD film after sudden stop. EHD film get collapsed in two stages—in the first stage FT rapidly falls whereas in the second stage the central FT remains nearly uniform and thus gets entrapped between the contact pair after the contact close. The EHD film is so stiff at the contact point that it does not get directly affected with the change in the loading.

Dowson and Hamrock [2], presented a paper on the Numerical solution of the Elastohydrodynamic Lubrication (EHL) for point contacts. They considered the geometry of the contacting solids for developing the equivalent radius of curvature equation. Then after they considered the point contact in the form of elliptical contact discussed about the significance of its asperity. Reynolds equation was used to show the EHL for point contact in one direction. They considered the FT calculation by considering the elastic deflection, geometry separation and the central FT due to hydrodynamic effects. After considering the boundary condition, they used the relaxation method to solve the EHL equation. Overall, the work was a remarkable basic for the elliptical contact problem. Dowson and Hamrock [3], presented a paper on the Numerical solution of the isothermal Elastohydrodynamic Lubrication (EHL) for point contacts having elliptical shape. In this paper they varied the asperity ratio of the elliptical contact by changing the configuration of approaching geometries. The work started from a ball to plate contact as point contact to a line contact problem. The asperity ratios were termed from 1 to 8 based on the type of contact. The use of the finding of this research one can easily map the contour map of FT over the area. A comparison was done between the proposed method and Archard-Cowing formula. The results were carried with other experiments also and an equation for pointy and line contact FT was developed.

Huges et al. [4], presented a paper on the coupling of elastohydrodynamic problem. The initiative here was to couple all the equations of Elastohydrodynamic Lubrication (EHL) to define the line contact of problem of EHL. The idea here was to develop a numerical solver for the study of a thin film and rough surface contact which are common in gears. Deformation behaviour of the surface was found to have a dominant effect. This method was applied for both Newtonian and non-Newtonian conditions with rough and smooth surfaces. The partial coupling method was also used and the results were compared to each other. It was found that for a line contact problem full coupling is preferable as the above partial coupling method. Along with these findings it was also found that the full coupling method needs a complete solution matrix and thus it was time consuming.

Huges et al. [5], presented a paper to determine the deflection due to pressure acting on a semi-infinite body by using differential deflection technique. This can be easily accommodated to the Elastohydrodynamic Lubrication (EHL) problem of line contact. This method does not use the full matrix penalty method but allows a fully coupled approach to be used. Differential deflection method specifies the deflection in a distributed implicit way instead of using the explicit way. This method releases the result very common to earlier results with different methods however the time consumption for the overall process was on the lower side as compared earlier. This was an advantage of this process as compared to the earlier used process. In order to ensure this the verification of this method was carried for accuracy of results and efficiency. Accuracy was accessed by comparing the results with traditional methods whereas the efficiency was compared with the time considered by traditional methods.

Evans and Snidle [6], presented a paper to describe a numerical method for solving point contact problems under moderate load in the Elastohydrodynamic Lubrication problem with isothermal conditions. From the basic equation of deformation, elastic

deformation equation was developed. The iterative method of solution was used with a finite difference grid method. Numerical analysis was used for a range of loads and speeds for lubricants with both exponential and power law viscosity-pressure relationship. Flow diagram for the Numerical solution was also developed. Results were generated for moving rough surfaces where the roughness of the surface was confirmed with a profilometer. Similar exercise was also performed for smooth surfaces. This exercise was performed for various values of slide to roll ratios with extremely low lambda value.

Evans and Hughes [7], presented a paper for deflection evaluation in semi-infinite bodies by using the differential method. This work presented a procedure for evaluating the Laplacian of deflection of the semi-infinite body subjected pressure loading. In this analysis both line and point contacts were considered. Expression for deflection due to line contact was developed from the basic equation. A quadrature mesh was used with defined spacing and the pressure to the corresponding points were also determined. An expression in relation with deflection and pressure was developed with the assumption that the distribution of pressure was continuous and it was also differentiable. On a similar way the Laplacian equation of the deflection at a point was also developed. Numerical solutions of both the line contact problem and point contact problem were evaluated and various charts were plotted. Finally Analytic expression for the second spatial derivative of deflection was developed for one dimensional pressure variation along with Laplacian of its deflection when subjected to two-dimensional pressure variation. The accuracy of this method was verified by using a hertzian pressure distribution for both point and Line contact. The method developed here helped in exploiting the benefits of full coupling in these problems.

Han et al. [8], presented a paper to evaluate the film parameter and friction coefficient of the helical gear after considering surface roughness and load variation. A load distribution model was employed here with finite element analysis with Geometric and kinematic parameter calculation at each point. The plane of action was considered on which load sharing concept was also utilized. Investigations and discussion were done for the effect of surface roughness, rotational speed, and transmitted load on the friction coefficient. A Flowchart for the calculation methodology was also developed. To verify the proposed model same data set was applied which was used in earlier similar research. For this research the approach used was the extended version of spur gears to helical gear. It was concluded that friction coefficient increases with rougher surface, increasing the pinion speed can also speed the transition period of Lubrication from boundary to mixed lubrication condition. Friction coefficients are directly proportional to the applied load. This method of approach does not need finite element analysis for load distribution. There is no need of solving the complete Elastohydrodynamic lubrication solution and the results are also reasonable which brings on the added advantages of this approach.

Kaneta and Cameron [9], presented a paper to study the effects of Asperities in Elastohydrodynamic Lubrication (EHL). In the experimental work optical profilometry was used for the study of rough surfaces under point contact in lubrication. In the experimental method the traction force was transmitted through the EHL film

to rotate the ball in race. In one part of the ball's surface asperities were sputtered. Sputtering material was chromium. A defined load of 4 kg which referred to certain Hertz pressure of 0.55Gpa was used. The applied force was sufficiently high so that the ball can slip but lower enough to avoid the asperity damage. Either ball or disc rotation was prevented to get pure sliding. The minimum oil thickness measurable was about 40 nm. One of the interesting finding of this experimentation was that the asperities were behaving in a different manner for rolling and sliding motion. FT in sliding was too low making it complex.

Li and Kahraman [10], presented a paper for a transient mixed Elastohydrodynamic lubrication (EHL) model for spur gears. The condition of lubricant was considered as non-Newtonian. EHL model developed in this research was applicable for spur gear and designed to follow continuously the tooth pair contact from root to tip. Gear load distribution method was used to predict the distribution of normal tooth force along the line of action. One dimensional transient Reynold's equation was used for the analysis. FT of the contact point included the gap between the two surfaces before and after the elastic deformation including the roughness of the surfaces. The domain for this research was limited from start of active profile to end of tip of the tooth gears. Minimum FT variation along the surface of the contact line was determined by using the EHL model.

Nijenbanning et al. [11], presented a paper for the evaluation of FT in Elastohydrodynamic lubricated elliptic contact. Solver used for circular contact was extended for elliptical contact cases. Available literatures were used to verify the new extended model for elliptical contact. Detailed computational analysis is done and the tendencies observed are directed back to the modeling equation. It is shown that from the line of contact analysis the pressure and the FT can be accurately predicted. For this the entrainment direction is considered as perpendicular to the major principal axis of the contact ellipse. A formula is also provided which can predict the central and minimum FT. The formula incorporates asymptotic behavior and is valid to all load condition.

Ranger [12], presented a paper for solution of elasto-hydrodynamic problem for the point contact. Two surfaces are considered in contact such that one is elastic sphere and the other is elastic plane. The elastic sphere is rolling on the elastic plane. Operating condition is considered as steady running with isothermal film. The pressure viscosity exponential characteristic is considered along with solving the Reynold's equation and elasticity equation simultaneously. Deformation matrix considered as sufficiently compact. To expand the solution the results were subjected to statistical methos which was multiple regression. The result coming out of this was well comparable with published data.

Chittenden et al. [13], presented a paper on theoretical analysis of the Isothermal Electrohydrodynamic Lubrication (EHL) of concentrated contacts. In this research the direction of lubricant entertainment was considered coincident with the major axis of the contact ellipse due to Hertzian pressure. This paper discusses about the significance of this research work with the earlier available data in terms of direction of entertainment of lubricant oil.

A total of seventy-two new EHL solutions was computed which got distributed in two papers published by him. Thirty-four of them are discussed in this paper. This paper considered the aspect ratio of the contact ellipse for the analysis. Analytical formulation was done from basic equations. Elastic deformation with lubricant properties were considered along with EHL equation. A computer program was developed, and a computer chart was developed for that.

Ai and Cheng [14], presented a paper on a transient Elastohydrodynamic lubrication (EHL) analysis for line contacts with measured surface roughness by using multigrid method. Reynolds equation was consider as the governing equation whereas the FT equation consists of the surface roughness effect also. Modified two slope exponential model was considered for Pressure viscosity relation. Euler's stability consideration was used for the discretization of Reynolds equation. It was found that surface roughness induces transient effect. This has a remarkable effect on pressure distribution and profile FT. With the increase in relative speed between the contacting surfaces pressure fluctuation also increases.

Wang et al. [15], has analyzed the relationship between the tribology and micropitting with different geometric slide to roll ratio by using two different lubricants in a test rig. To measure the level of micropitting an optical microscope was used. During the performing this research humidity was also considered. Result of this research shows a relationship of humidity and slide to roll ratio as with increase in humidity the higher wear, Whereas the Friction change has no such huge effect. But the micropitting observed was due to combination of both humidity and friction.

Robin et al. [16] reviewed a case study of ISO/TS 6336–22 micropitting calculation. ISO/TS 6336–22 refers to a method of analyzing the micropitting load capacity of external spur gear sets based on testing. It considers the relationship between the micropitting and minimum FT such that its ratio with the permissible specific lubricant FT is the safety factor against micropitting, considered as lambda ratio. Analysis case was performed. Analytical calculation using method B was used. Other analysis were also performed using wind turbine gear set. Surface topography was analyzed after testing by taking various images of the experimental gears. Micropitting was predicted on the surface of tooth and thus it was concluded that micropitting is not projected by FT and surface roughness alone.

Xu et al. [17] analyzed the tooth profile modification method of spur gears for reduction in micropitting. Hertz contact stress points were shown on tooth flank by meshing the gear teeth. Tip relief methodology is used here and the corresponding relations in mathematical equations were developed. The micropitting was considered to be related to hertz stresses and the result in tooth curvature modification the hertz stress value at critical points were reduced thus, considering the reduction in micropitting.

Redlich et al. [18], analyzed the mixed lubrication regime with the Elastohydrodynamic lubrication model. Point contacts were considered for the detailed analysis. Lubricant film pressure and contact pressures were calculated for such regime and plotted. Surface roughness was also considered in the picture and hence to solve the equations iterations were required. Multigrid approach was used to solve the problem. The analysis was further extended to study the effect of different rough surfaces with

some velocities. The velocities include a pure sliding condition to different surface velocities. Goodyer et al. [19], analyzed the multigrid approach for solving the Elastohydrodynamic lubrication model. Because of the requirement of high iteration for solving the Elastohydrodynamic model, the multigrid approach have been proven very useful. The convergence of the equation in order to get the results were very good for steady problems. The cavitation was also found as the responsible point in the holding of convergence of the equation. This research basically applied to both steady and transient conditions and explained the way the equations were getting converged to get the solution.

Jolkin and Larsson [20], analyzed the oil FT determination with ball and disc arrangement using the optical method as well to calculate the Elastohydrodynamic lubricated FT. Polyalphaolefin synthetic oil was used in the experimentation. A numerical method was also parallelly used to define the oil FT and the pressure envelop was also compared. Both results were showing good agreement for a given pressure and temperature conditions. The accuracy of the results were dependent on the calculation accuracy and measurement accuracy as well. Khalilarya et al. [21], analyzed the numerical model of Elastohydrodynamic Lubrication for involute spur gear. To prepare the model, the lubricant was considered to be following the Newton's law. The whole analysis was performed on the line of contact. A numerical model was prepared by considering the Gear tooth bending shearing, compression and tilting activity and it was very much in line with analytical and experimental results. The output of this experiment was in the form of the results available for various values of lubricant FT and pressure distribution along with pressure envelop. Chen et al. [22], analyzed the numerical solution between two surfaces having different roughness with high roughness value. Some rough particles were also included between these surfaces to determine the effect. Reynold's equation was used for the film pressure distribution. Both the roughness and third particles affected the results. Non dimensional parameters were considered for the detailed analysis. Various conditions were analyzed and it was found that the oil thickness was diminishing fast because of the rough surface and presence of particles in between them.

### **Outcome of Literature Survey**

Many papers were studied for this research work, however top fourteen paper's analyses were presented in this work which are useful for further step analysis. The findings are depicted as below:

- A good amount of research work database is there which is helpful in developing the Elastohydrodynamic lubrication (EHL) equation.
- Various studies inform various methodologies for the solution of EHL problem.
- A good methodology is available for point, circular and elliptical contact EHL analysis.
- Various data are available for spur gear contact EHL analysis.
- Contact line is considered as the typical area for the EHL analysis.
- Method to transform the relations used for the circular contact point to the elliptical contact point is also provided in numerous papers.

- Various useful methods for experimental determination of fluid FT is explained in a detailed way with most of them in practical use.
- Data related to Helical gear contact line EHL analysis is not abundant.
- It was clearly found that the relative speed, load variation and surface roughness are the key parameters in lubricant thickness.

## **1.5 Problem Definition**

A pair of helical gear is considered for this research work such that the no of teeth on Pinion is 18, and the number of teeth on gear is 58, Helix angle is 15 degrees, pressure angle is 20 degrees. Pinion is rotating at 1470 rpm with input power 0.5Hp. An Elastohydrodynamic lubrication (EHL) model need to be developed for the isothermal condition and Newtonian lubricant needs to be considered.

The model needs to calculate the FT on the contact line at critical points as discussed in Sect. 1.2. The input shaft pinion is mounted on the cantilever shaft whereas the output pair is mounted on a simply supported condition. The corresponding value needs to be reviewed in KISSOFT.

## **1.6 Software Packages in Use**

For solving the EHL Model Mathcad prime-8 a local inhouse developed tool will be used. Kissoft will be used to plot various plots to compare the result of the model.

## **1.7 Elastohydrodynamic Lubrication (EHL) Model Development**

EHL model consists of two basic equations, one belonging to elasticity theory for deformations and the other related to the fluid film hydrodynamic equations. In this section we will be discussing about point contact problematic with related equations and discuss the discretization of these equations by using the Finite difference technique.

### **Hydrodynamic Equation**

Reynolds equation is used here as the hydrodynamic equation which can be displayed for a 2d form as below:

$$\frac{\partial}{\partial x} \left( \sigma_x \frac{\partial p}{\partial x} \right) + \frac{\partial}{\partial y} \left( \sigma_y \frac{\partial p}{\partial y} \right) - \frac{\partial}{\partial x} (\rho \bar{u} h) - \frac{\partial}{\partial y} (\rho \bar{v} h) - \frac{\partial}{\partial t} (\rho h) = 0 \quad (16)$$

where,

$$\bar{u} = \frac{u_1 + u_2}{2}$$

$$\bar{v} = \frac{v_1 + v_2}{2}$$

For the case of Newtonian oil behaviour, the flow factors can be represented as:

$$\sigma_x = \sigma_y = \frac{\rho h^3}{12\eta} \quad (17)$$

### The Elastic deformation equation

Here the point of interest is to determine the gap between the two contacting surfaces. The approximate value of the gap can be known by using equivalent surface concept as discussed in Sect. 1.3. Considering the oil entrant direction as x, the thickness of oil can be written as below:

$$h(x, y) = h_u(x, y) + d(x, y) + h_0 \quad (18)$$

$h_u(x, y)$  is representing the undeformed gap,  $d(x, y)$  is the elastic deformation and  $h_0$  is the constant that shows the distance of approach. Undeformed gap can be calculated as shown in Sect. 1.3. Timoshenko and Goodier provided the methodology to determine the elastic deformation under pressure loading,  $p(x,y)$  as:

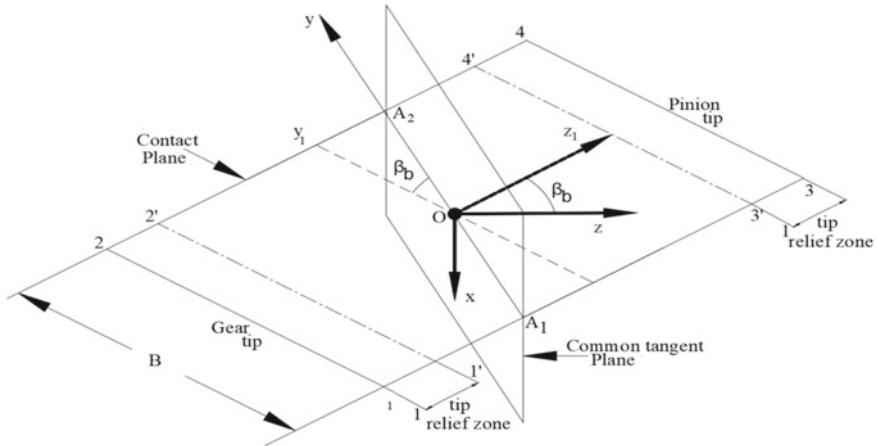
$$d(x, y) = \frac{2}{\Pi E'} \iint \frac{p(x, y)}{\sqrt{(x - x_1)^2 + (y - y_1)^2}} dx dy \quad (19)$$

where  $x_1, y_1$  represent the point's coordinate where deflection is supposed to be calculated and

$$\frac{2}{E'} = \frac{1 - v_1^2}{E_1} + \frac{1 - v_2^2}{E_2}$$

In the above figure A<sub>1</sub>A<sub>2</sub> shows the contact line for the helical gear in the contact plane. Point O shows the midpoint of face width. Oz<sub>1</sub> represents the common normal for the helical gear teeth and the common tangent is shown by the Oxy plane. Spur gear's contact gap is minimum in the direction Oz<sub>1</sub>. For the case of the helical gear pair, the lubricant film growth depends on the space between them in the standard direction, Oz, & depends on the motion of the surfaces comparative to the line of contact. This comparative motion takes place in the Ox direction (Fig. 5).

Figure 6 shows the two curves to calculate the gap at the desired point p<sub>1</sub>(x,y,z) by considering the normal, tangent and transverse plane associated with the geometry. Figure 6b and c are showing the detailed view for performing the calculation.P<sub>1</sub>B



**Fig. 5** Plane of action and various forces

dimension as shown in the Fig. 6c is the gap in the transverse plane and can be obtained as:

$$P_1B = z' = \frac{x^2}{2R_1'(y_{p1}')} \quad (20)$$

$\beta_b$  shows the inclination of the normal plane to the transverse plane. So, the gap referring to the concerned point is given by distance  $P_1A$ , as:

$$P_1A = z_1 = z_1/\cos\beta_b \quad (21)$$

Substituting Eqs. (20) in (21) gives:

$$z_1 = \frac{x^2 \cos\beta_b}{2R_1'(y_{p1}')}.$$

This equation gives the relation between  $z_1$  and  $x$ .

From Fig. 6c we have:

$$y_{p1}' = y_p + z_1 \tan\beta_b$$

It is very clear from above that there are corresponding points on each curve from which a transverse plane gap can be determined.

So, the radius  $R_1'$  is dependent on the  $x$ ,  $y$  and  $z$  coordinates for the point on curve 1. Contact geometry for the corresponding point is given as:

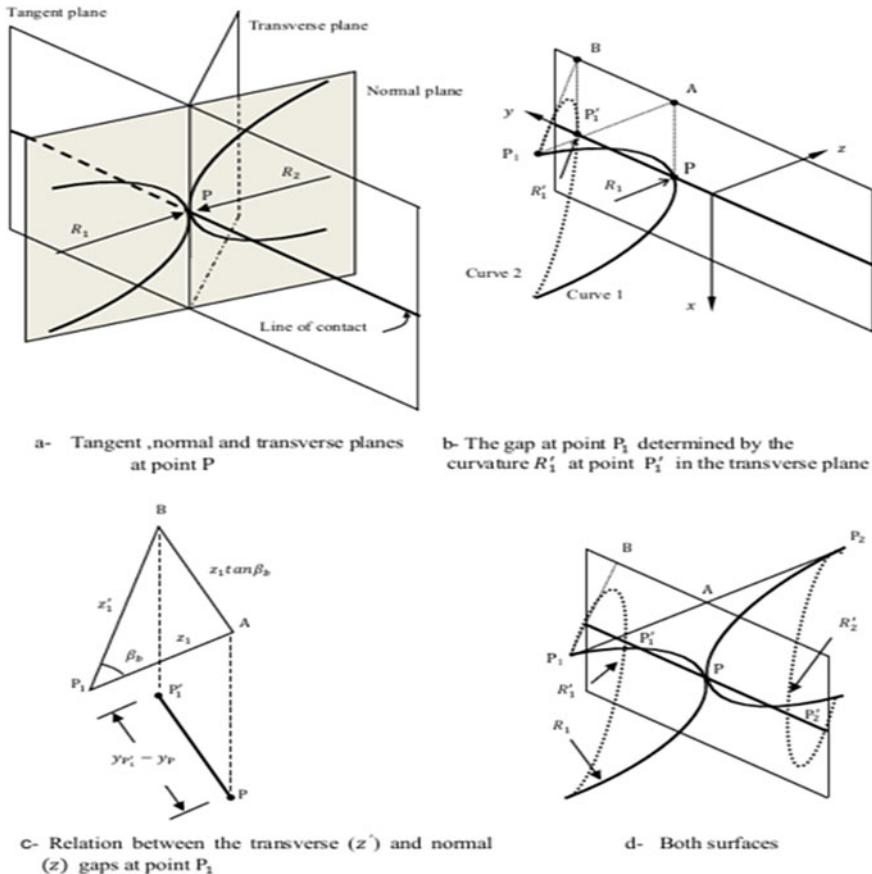


Fig. 6 Contact Geometry of helical gear teeth

$$z_1(x, y) = \frac{x^2 \cos \beta_b}{2R_1'(y + z_1 \tan \beta_b)} \quad (22)$$

From the above equation it is very clear that for a given point to solve this equation one needs to go for the convergence of the values, and this can be obtained by modifying the argument  $2R_1'(y + z_1 \tan \beta_b)$  with successive  $z_1$  values.

Another equation can be calculated for the other geometry as:

$$y_{P2'} = y_p - z_2 \tan \beta_b$$

The gap  $z_2$  refers to opposite side gap calculated by:

$$z_2(x, y) = \frac{x^2 \cos \beta_b}{2R_2'(y - z_2 \tan \beta_b)} \quad (23)$$

In the normal direction the net undeformed gap,  $h_u$ , in the normal direction is the addition of  $z_1$  and  $z_2$ , so we have

$$h_u = z_1 + z_2$$

### The Pressure viscosity relation

From Eq. (24) we get an empirical relation between viscosity and Pressure. This is considered in this work in the following form (Clarke et al. 2006)

$$\eta = \eta_0 \exp \left\{ \ln \left( \frac{\eta_0}{\kappa} \right) (1 + \chi p)^2 - 1 \right\} \quad (24)$$

where,

$$\kappa = 63.15 \times 10^{-6} \text{ Pa.s}$$

$$\chi = 5.1 \text{ GPa}^{-1} \text{ Pa.s}$$

$$Z = \frac{\alpha_0}{\chi \ln \left( \frac{\eta_0}{\kappa} \right)}$$

The Pressure density relation.

Per Dowson and Higginson (1966) the relation is considered as below:

$$\rho = \rho(\rho) = \rho_0 \left( \frac{1 + \chi p}{1 + \lambda p} \right) \quad (25)$$

Values of constant for the pressure coefficients are as below:

$$\gamma = 2.266 \text{ GPa}^{-1} \text{ and } \lambda = 1.683 \text{ GPa}^{-1}.$$

### 1.8 Surface Elastic Deflection-Evaluation

As per Eq. (18) we know that we need to evaluate the surface deflection to determine the FT in the area of interest. It is evaluated from Eq. (19).

One more method to solve this is to use a numerical method for any general pressure distribution occurring between the contact of rough surfaces. This integral is transformed to a summation of influence coefficients multiplied by the pressure value for all the points in the solution domain. This is because of numerical discretization. This can be written (Evans and Hughes 2000) as:

$$d(x_i, y_j) = \frac{2}{\Pi E'} \sum_{K=1}^N \sum_{L=1}^M g_{k-i,l-j} p_{k,l} \quad (26)$$

## Discretization Method

Under this topic, there is a brief description of solution for the steady state point contact EHL problem by using the Finite Difference method.

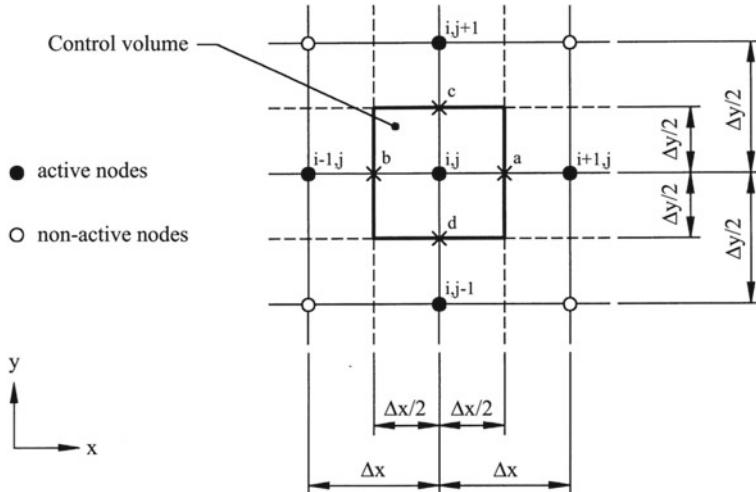
Discretization of the Poiseuille flow term of the Reynolds equation is done by using the central difference method, whereas the other term is discretized using the backward difference method.

Discretization process is carried out by using a rectangular control volume centred about each node, such that the dimension of the control volume is  $\Delta x$ ,  $\Delta y$ . Figure 7 shows a single control volume used to represent the Poiseuille term for node  $i, j$ .

Considering the illustration given in Fig. 8, Poiseuille term, which involves the flow in  $x$  and  $y$  direction due to pressure gradient, can be shown as:

$$\frac{\partial}{\partial x} \left( \sigma_x \frac{\partial p}{\partial x} \right) + \frac{\partial}{\partial y} \left( \sigma_y \frac{\partial p}{\partial y} \right) = \frac{\sigma_a \left( \frac{\partial p}{\partial x} \right)_a - \sigma_b \left( \frac{\partial p}{\partial x} \right)_b}{\Delta x} + \frac{\sigma_c \left( \frac{\partial p}{\partial x} \right)_c - \sigma_d \left( \frac{\partial p}{\partial x} \right)_d}{\Delta y} \quad (27)$$

Here mean values are considered at the boundaries of the control volume for the flow factors given by



**Fig. 7** Control volume representations taken from Holmes (2002)

$$\begin{aligned}\sigma_a &= \frac{(\sigma_{i+1,j} + \sigma_{i,j})}{2}, \sigma_b = \frac{(\sigma_{i,j} + \sigma_{i-1,j})}{2} \\ \sigma_c &= \frac{(\sigma_{i,j+1} + \sigma_{i,j})}{2}, \sigma_d = \frac{(\sigma_{i,j} + \sigma_{i,j-1})}{2}\end{aligned}\quad (28)$$

For the pressure gradient the central difference approximation gives:

$$\begin{aligned}\left(\frac{\partial p}{\partial x}\right)_a &= \frac{(p_{i+1,j} - p_{i,j})}{\Delta x}, \left(\frac{\partial p}{\partial x}\right)_b = \frac{(p_{i,j} - p_{i-1,j})}{\Delta x} \\ \left(\frac{\partial p}{\partial y}\right)_c &= \frac{(p_{i,j+1} - p_{i,j})}{\Delta y}, \left(\frac{\partial p}{\partial y}\right)_d = \frac{(p_{i,j} - p_{i,j-1})}{\Delta y}\end{aligned}\quad (29)$$

Wedge term can be shown as

$$-\bar{u}\left(\rho \frac{\partial h}{\partial x} + h \frac{\partial \rho}{\partial p} \frac{\partial p}{\partial x}\right) - \bar{v}\left(\rho \frac{\partial h}{\partial y} + h \frac{\partial \rho}{\partial p} \frac{\partial p}{\partial y}\right)$$

Same can be represented in the central difference approximation method as

$$\begin{aligned}-\bar{u}h \frac{(p_{i+1,j} - p_{i-1,j})}{2\Delta x} - \bar{u}\rho \frac{(h_{i+1,j} - h_{i-1,j})}{2\Delta x} \\ -\bar{v}h \frac{\partial \rho}{\partial p} \frac{(p_{i,j+1} - p_{i,j-1})}{2\Delta y} - \bar{v}\rho \frac{(h_{i,j+1} - h_{i,j-1})}{2\Delta y}\end{aligned}\quad (30)$$

The values  $\rho, h\left(\frac{\partial \rho}{\partial p}\right), \bar{u}, \bar{v}$  in this equation are taken at the node where the hydrodynamic equation is applied. Finally the equation can be obtained as:

$$\begin{aligned}\sigma_a \left( \frac{p_{i+1,j} - p_{i,j}}{\Delta x^2} \right) - \sigma_b \left( \frac{p_{i,j} - p_{i-1,j}}{\Delta x^2} \right) + \sigma_c \left( \frac{p_{i,j+1} - p_{i,j}}{\Delta y^2} \right) - \sigma_d \left( \frac{p_{i,j} - p_{i,j-1}}{\Delta y^2} \right) \\ -\bar{u}h \frac{(p_{i+1,j} - p_{i-1,j})}{2\Delta x} - \bar{u}\rho \frac{(h_{i+1,j} - h_{i-1,j})}{2\Delta x} \\ -\bar{v}h \frac{\partial \rho}{\partial p} \frac{(p_{i,j+1} - p_{i,j-1})}{2\Delta y} - \bar{v}\rho \frac{(h_{i,j+1} - h_{i,j-1})}{2\Delta y} = 0\end{aligned}$$

## 1.9 Formation of Deformation Equation

This is formed after combining Eqs. (26) in (18) as :

$$h(x, y) = h_u(x, y) + \frac{2}{\Pi E'} \sum_{K=1}^N \sum_{L=1}^M g_{k-i,l-j} p_{k,l} + h_0 \quad (31)$$

The equation can also be mentioned as bow:

$$\nabla^2 h = \nabla^2 h_u + \frac{2}{\Pi E'} \sum_{K=1}^N \sum_{L=1}^M g_{k-i,l-j} p_{k,l} \quad (32)$$

where,

$$\nabla^2 h = \frac{\partial^2 h}{\partial x^2} + \frac{\partial^2 h}{\partial y^2}$$

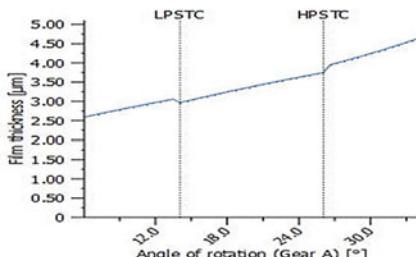
## 1.10 Results and Discussions

For the given problem state the EHL model was executed. A local developed tool in MathCad Prime is used to plot the results. A comparison chart was plotted with Kisssoft data under various conditions as shown below:

- a. Input stage: Lubricant thickness as per Kisssoft result (left) and EHL model result (right).
- b. Output stage: Lubricant thickness as per Kisssoft result (left) and EHL model result (right) (Figs. 8 and 9).

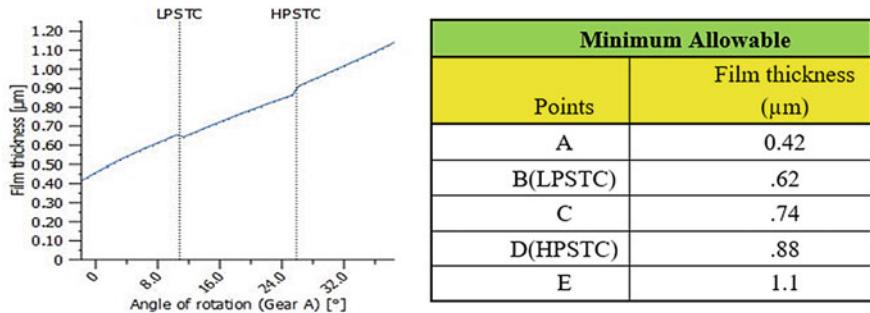
### Discussion of the Results

- The values of output at various points for the EHL model and Kisssoft output is comparable and is in good accuracy for the output shaft pair. The accuracy lies in the range of 5%.
- There is a good amount of variation in the EHL model result and Kisssoft result and the difference is in the range of 20–32%. This is a big margin. One possible reason for the difference would be that deflection of the input pinion shaft itself which is cantilever in this case.



Minimum Allowable	
Points	Film thickness (μm)
A	2.2
B(LPSTC)	2.9
C	3.1
D(HPSTC)	3.4
E	4.3

Fig. 8 Kisssoft result (left) and MathCad prime result (right)



**Fig. 9** Kisssoft result (left) and MathCad prime result (right)

- Deflection of the shaft in the case of output stage is negligible however if the value of shaft deflection would have been considered then a more accurate model could have been produced.
- From the available literature review it was clearly observed that with the application of load the fluid FT was also decreasing and the same is observed in the output stage pair also where the load is increasing, and the fluid FT is also decreasing as compared to the input stage for the same environment.
- Variation of FT along the path of contact and its nature is very much in line with the research data available, like its characteristics with speed and load.

### 1.11 Conclusion and Future Scope

EHL model prepared for the subject case is accurate as the data coming out of the EHL model looks to have a good agreement with the Kisssoft result. It was also found that the film thickness was considerable at the load changing points. Variation of the lubricant film thickness over the contact line looks very much dependent on the contact pressure.

A good future scope could be the analysis of the same for non-Newtonian fluid considering the temperature effect. It is also important to consider the external effects which are causing more deflections in the meshing pair resulting in the increased value of oil film thickness.

### References

1. Glovnea RP (2009) Transient phenomena in elastohydrodynamic lubrication. Res Signpost 37(661):227–262
2. Dowson D, Hamrock BJ (1976a) Isothermal elastohydrodynamic lubrication point contact part-I. J Lubr Technol:223–228
3. Dowson D, Hamrock BJ (1976b) Isothermal elastohydrodynamic lubrication point contact part-II. J Lubr Technol:118

4. Huges TG, Elcote CD, Evans HP (1997) On the coupling of the elastohydrodynamic problem. *Process Instit Mech Eng* 2012(part c)
5. Huges TG, Elcote CD, Evans HP (1999) Coupled solution of the elastohydrodynamic line contact using a differential deflection method. *Process Instit Mech Eng* 2014(part C)
6. Evans HP, Snidle RW (1981) The isothermal elstohydrodynamic lubrication of spheres. *J Lubr Technol* 103:547–557
7. Evans HP, Hughes TG (2015) Evaluation of deflection of sem infinite bodies by differential method. *Process Instit Mech Eng*:214
8. Han L, Zhang D-W, Wang F-J (2013) Predicting film parameter and friction coefficient of helical gears considering surface roughness and load variation. *Tribology* 56(1):49–57
9. Kaneta M, Cameron A (1980) Effect of asperities in elastohydrodynamic lubrication. *Tribology* 102:374–378
10. Li S, Kahraman A (2010) A transient mixed elastohydrodynamic lubrication model for spur gear pairs. *J Tribol*:132
11. Nijenbanning G, Venner CH, Moes H (1994) Film thickness in elastohydrodynamically lubricated elliptic contacts. *J Wear* 176:217–229
12. Ranger AP (1975) The solution of the point contact elasto-hydrodynamic problem. *Int J R Soc* 346:227–244
13. Chittenden RJ, Dowson D, Dunn JF, Taylor CM (1985) A theoretical analysis of isothermal elastohydrodynamic lubrication of concentrated contacts. *J R Soc* 397:245–269
14. Ai X, Cheng HS (1994) A transient EHL analysis for line contacts with measured surface roughness using multigrid technique. *J Tribol* 116:549–556
15. Wang X, Dong Y, zhang J, Bal D (2022) On the relationship between micropitting and friction/ wear of rolling-sliding steel/steel contacts. *ACM J*:64–71
16. Robin O, Michaud M, Keller J (2021) A Case study of ISO/TS 6336-22 micropitting calculation. *J Gear Technol* 11:48–57
17. Xu X, Liang Y, Wang Z (2022) A Novel tooth relief method for reducing micro-pitting of spur gears. *ACM J*:1–34
18. Redlich AC, Bartel D, Schorr H, Deters L (2000) A deterministic EHL model for point contacts in mixed lubrication regime. *Thinning Films Tribol Interfaces*:85–93
19. Goodyer CE, Fairlie R, Berzins M, Scales LE (2000) An in depth investigation of the multigrid approach to steady and Transient EHL problems. *Thinning Films Tribol Interfaces* 38:95–102
20. Jolkin A, Larsson R (2000) Determination of Lubricant compressibility in EHL conjuncions using the Hybrid technique. *Thinning Films Tribol Interfaces* 38:589–596
21. Khalilarya S, Mirzaee I, Mohammadpour M, Vahid DJ (2010) A numerical model to calculate Elastohydrodynamic (EHL) properties in involute spur gears. In: International conference on mechanical and electrical technology (ICMET 2010)
22. Chen K, Zeng L, Chen J, Ding X (2020) Analysis of line contact with the particles under rougfh contact surface. In: *Advances in material science and engineering*, vol 2020

# ABC Algorithm for Evaluating the Performance of the SVC and Optimal Power Flow



N. Kalpana

**Abstract** In this work, the Artificial Bee Colony (ABC) approach is investigated as a potential solution for optimum power flow (OPF) issues that arise in power systems through the utilisation of a static VAR compensator (SVC). By utilising SVC devices and making use of ABC, the system is able to bring down the overall generating costs of a power system, hence saving money. Also, it helps keep the voltage steady. The ABC is established predicated mostly on honey bees hunting activity in order to identify the proper nectars. The algorithm is a recently created optimization tool for use in power systems. To measure how well the proposed ABC approach performed, it was juxtaposed to other renowned optimization methods on IEEE 30-bus & 11-bus systems. The findings indicate that ABC can be confidently accepted as a method for dealing with nonlinear issues in power systems; in fact, this method is extensively utilised in power systems. The results suggest that ABC may be strongly accepted for dealing with nonlinear problems in power systems, and it is frequently employed in power systems.

**Keywords** Optimization · Static VAR compensator (SVC) · Artificial bee colony · Optimal power flow · Continuous power flow · Voltage stability active-power losses

## 1 Introduction

Over the past couple of generations, there has been a tremendous expansion of electric power networks. A nation's economy and rapid industrialization depend heavily on energy, which is available at power stations and then transfers via HV cables and substations to final users. Power plant operations have proven challenging due to the need for a new electrical system to respond to changing load demands and provide high-quality electricity. Because of the intense pressures placed on these systems, such as unexpected power flows of power and significant losses, these systems are constantly challenged, and the power grid has to deal with voltage instability so

---

N. Kalpana (✉)  
DEEE, MECS, Hyderabad, India  
e-mail: [kalpana@matrusri.edu.in](mailto:kalpana@matrusri.edu.in)

that all load buses get the voltage they need. When these conditions aren't met, the voltage drops and instability will ensue. Therefore, further developing the voltage dependability edge of power systems is significant. In 1988, Hingorani [1] created FACTS in an effort to overcome the aforementioned difficulties and optimise the use of power system equipment. This means that transmission line capacities and power system security have both seen major boosts owing to FACTS devices and their associated circuits.

SVCs, familiar FACTS devices, consume or generate bus reactive power to stabilise voltage depending on its location. When it comes to the planning and execution of electrical systems, OPF is an indispensable factor. OPF is essential from the perspective of the setup and functioning of electrical-systems. In general, System data, which is a sizable multidimensional, nonlinear dataset, is employed for handling OPF. It becomes even more difficult when variable restrictions and FACTS device settings are incorporated. To manage convergence to the optimal solution, a number of analytical and conventional optimization techniques are employed, EX: the extended reduced gradient technique, the Newton–Raphson technique, the interior point approach and the quadratic programming method, among others. For traditional optimization methods to avoid being caught in local minima, it is necessary to start from a position that is relatively close to the solution. The standard of the responses becomes extremely dependent on the starting values as the number of problem parameters grows. These traditional approaches have their limitations, especially with the rise of computational technologies, computer-based methods are now preferred. Over the past few decades, there has been a sharp increase in interest in applying heuristic optimization techniques to problems with the power system. The heuristic optimization techniques employ probabilistic transition rules as opposed to deterministic ones. avoid using information that is derived from other things, possess the capacity to avoid getting trapped in a regional minimum, and be able to deal with large-scale nonlinear issues. In recent years, some of the most well-known heuristics, like NSPSO,PSO and DE [2] are put to use in an effort to minimise the total cost of generation while also maintaining load bus voltages within the parameters. This was accomplished by locating the areas in the best way possible of SVC devices and the values of their parameters. In power systems, both heuristic optimization algorithms and conventional optimization techniques have been applied, each with its own advantages and disadvantages.

An optimization approach that is named the artificial bee colony (ABC) method is given in here as a way to address the OPF problem. This algorithm was modelled after the way honey bee swarms search for food, which served as the inspiration for developing this method. Further, the power system is viewed as combined with SVC susceptance concept, and OPF based on ABC is used to address the resulting problem. Under the stated max.power limitations, generation constraints of active-power values, and bus voltage limits, the total generating cost is minimised. As a way to address the OPF issue, the factors of FACTS were found by applying ABC for both the IEEE 30 & 11-bus systems. In order to gauge how well this technique performs, it is compared to other optimization strategies. Like the second order gradient method, the reduced Hessian method, results in differential evolution.

The framework of the paper is divided into the following subsections: Under the second section, the OPF problem was formulated and the modelling of the power system explained. In Sect. 3 we discuss the ABC algorithm and how it was applied to OPF. Section 4 offers the results, while Sect. 5 conveys the discussion and conclusions. A brief description of the results is given in Sect. 4, followed by a discussion and conclusions in Sect. 5.

## 2 The Formulation of the OPF's Problems

### A SVC Model

The OPF used in this work's SVC susceptibility model, which was created for it and the load buses are regarded as being fed with reactive power [3], The SVC susceptance model shown in Fig. 1, [4–7] which accounts for MVAr injections at load buses, [8–10] was employed for the OPF used in this study [3]. When it comes to managing characteristics like load bus voltages, this parallel-linked SVC's output is made alternate between inductive and capacitive current.

For the bus r(Q<sub>r</sub>), the MVAr consumption by the SVC (Q<sub>sv</sub>) can be symbolised as

$$Q_{SV} = Q_r = -V_k^2 B_{SVC}$$

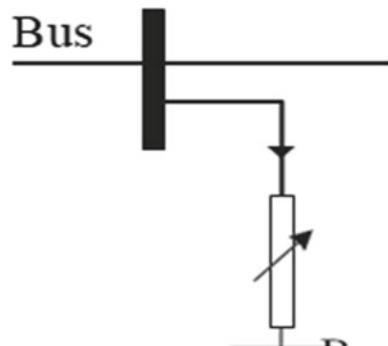
where V<sub>k</sub>—voltage on bus k

B<sub>SVC</sub>—bus-to-source voltage converter's equivalent susceptance.

### B. Optimal power flow

The OPF is a constrained optimisation issue with a constrained nonlinear objective function. The main aim is to optimise the whole generating cost by identifying the appropriate power values of production segments and the specifications of SVCs in the OPF problem considered here [11].

**Fig. 1** SVC susceptibility model



Here is how the OPF can be mathematically laid out:

$$\text{Mini (whole generation cost } f(x, u)) = \sum_{i=1}^{N_g} a_i + b_i P_{Gi} + C_i P_{Gi}^2.$$

subject to  $\begin{cases} m(s, c) = 0 \\ n(s, c) \leq 0 \end{cases}$

Here

$$s = [P_{G\text{Slack}} V_{LL} Q_G] = \text{State variables}$$

$$c = [P_G V Q_G Q_{SV}] = \text{Control variables}$$

$P_{G\text{slack}}$  = The slack bus's power

$V_{LL}$  = load bus voltage

$Q_{G0}$  = MVAr generation

$P_G$  = real power

$V_G$  = the generator voltage

$Q_{SV}$  = SVC MVAr

In order to achieve an OPF, it is necessary to lower the overall generating cost, which is denoted by the expression  $f(x, u)$ .

$$f = \sum_{i=1}^{N_h} a_i + b_i P_{Gi} + C_i P_{Gi}^2$$

here

$N_h$  no. of generators

$P_{Gi}$  real power generation at bus i;

$P_{Gi} - P_{Di}$  = losses

$$P_{Gi} - P_{Di} - \sum_{j=1}^{Nb} |V_i| |V_j| |Y_{ij}| \cos(\theta_{ij} + \delta_i + \delta_j) + P_{injeSvCi} = 0 \quad (3a)$$

$$Q_{Gi} - Q_{Di} - \sum_{j=1}^{Nb} |V_i| |V_j| |Y_{ij}| \cos(\theta_{ij} + \delta_i + \delta_j) + Q_{injeSvCi} = 0, 3 \quad (3b)$$

Above equations are the load flow equations  $m(s, c)$ .

$Q_{Gi}$  = the generation reactive-power.

$P_{Da}$  &  $Q_{Da}$  = Ath bus load demands,

$P_{injeSvCa}$  and  $Q_{injeSvCa}$  = Ath bus injected powers.

$$V_{Gi}^{\min} \leq V_{Gi} \leq V_{Gi}^{\max} = 1, \dots, N_a \quad (4a)$$

$$P_{Gi}^{\min} \leq P_{Gi} \leq P_{Gi}^{\max} = 1, \dots, N_a \quad (4b)$$

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max} = 1, \dots, N_a \quad (4c)$$

$$Q_{GiSVCi}^{\min} \leq Q_{SVCiGi} \leq Q_{GiSVCi}^{\max} = 1, \dots, N_{SVC} \quad (4d)$$

where  $N_{SVC}$ —SVC devices number.

### C. Sensitivity analysis

To deploy SVC in the proper locations, a power system sensitivity analysis was conducted and also to identify the bus number most sensitive to variations in reactive-power. Voltage stability significantly improved by the use of shunt compensation and in order to attain the highest possible level of efficiency, conducting a V–Q sensitivity analysis is necessary before deciding where to place SVC devices.

The Jacobian matrix for the power system was utilised for carrying out the sensitivity analysis. Variation in J-matrix elements and reactive power equals a differential in voltage value:

$$\Delta u = J_r^{-1} \nabla q \quad (5)$$

here  $J_R J_4 - J_3 J^{-1} J_2$  = system's reduced jacobian matrix.

## 3 The Artificial Bee Colony Algorithm Description

Nonlinear and multidimensional optimization problems have attracted a plenty of focus and consideration in science and engineering in couple years, and swarm intelligence-based optimization techniques have been at the centre of this attention. An innovative heuristic optimization strategy, the ABC [12–16] is used to identify optimal convergent solutions to optimization issues by mimicking swarms of foraging honey bees. In 2005, Karaboga developed the ABC algorithm. It was first seen while studying the behaviour of real bees as they foraged for and distributed honey. i.e. ABC adapted its foraging strategy from those of swarms of honey bees.

Both self-organization and the division of work are cornerstone concepts here. The labor(bees) of the ABC algorithm is distributed across three classes: workers(employers), observers(onlookers), and scouts [17–19].

As employed bees do, workers keep their attention fixed on a nearby source of nectar so that they don't forget where it came from. While scout bees are subjected to a rigorous degree of calculation, onlooker bees are responsible for compiling

information from worker bees and making a resource decision regarding nectar harvesting.

An explanation of the algorithm used in this study is provided below:

1. Firstly, prepare the ABC food supplies.
2. Calculate the power-flow using the N-R method in a campout starting with the given data.
3. Third, get the best function from the power flow solution. Choose the most viable option for initial sustenance. For  $k = 1$ , set the iteration counter.
4. Locate positions for currently employed bees
5. Utilizing the N-R method, provide a solution to the power flow issue by taking into account the positions of the workers bees.
6. The sixth step is to compute the new fitness function based on the relocated workers.
7. By contrasting the modified fitness function with the initialised one, we can learn which value is best.
8. Update the position of the observers to reflect their newfound proximity to food sources and proceed to Step 5.
9. Identify unimproved food sources when the “limit” is reached.
10. Assign the unaltered food sources found to the scout bees, find the power-flow problem, and then compute the fitness function.
11. Enhance the values of the best function that correspond as the best possible place for the food source.
12.  $K = K + 1$  and repeat from step 4 if  $K$  is less than the maximum number of repetitions.

### **ABC implementation to address the OPF problem [20]**

1. Specify ABC parameters. Details such as voltages of line and bus, real and reactive power, and SVC susceptance data are included.
2. arbitrary selection of the first parent vectors Eq. (6)

$$P_{mn} = P_{n \min} + \text{rand} X (P_{n \max} - P_{n \min})$$

$m = \text{index number}, n = \text{dimension number}$  (6)

3. Using the equation, determine the numbers for the fuel cost, as well as the FV.

$$FV(i) = \begin{cases} 1/(1 + FV_i) & 0 \leq f_i \\ 1 + \text{abs}(FVi) & 0 > f_i \end{cases} \quad (7)$$

4. Establish a fresh objective that is connected to below Eq (8). Through the utilisation of optimal power flow, one must determine the FV and fuel cost associated with each target vector. Determine which vector has the highest fitness rating and save that one.

$$R_{ij\text{new}} = R_{ij} + \emptyset \times (R_{ij} - R_{kj})(i, k, \emptyset = \text{rand}[-1, 1]) \quad (8)$$

5. Estimate the possibility of the fuel cost for every food source connected to a certain rule. To meet below Eq. (9), develop a different vector; check FV and pick the best (top) possibilities for each food source.

$$P_i = \frac{\text{fitness}_i}{\sum_{i=1}^{SN} \text{fitness}_i} \quad (9)$$

6. Drop the current target, choose a new one, and move on to Step 2 if the target wasn't enhanced or improved after the user-specified maximum number of tries.  
 7. The best answer that has been identified up to this point will be updated, and the iteration counter will advance by one.  
 8. When all of the end criteria have been met, the iteration process should be finished. In any other case, move on to Step 4 and carry out the iteration method as many times as necessary until the end requirements are met or the iteration counter reaches its maximum number.  
 9. Specifications of ABC method

Quantity of bees = 20  
 Nectars = 20  
 max. iterations = 500.

## 4 Test Results

### Case 1 IEEE 11-bus test systems

The Table 1 provides information about the IEEE 11 bus-system.

In addition to further assess the efficacy of the ABC technique, we apply differential evolution to the same problem. The optimised values of all powers, voltage angle, and total generating cost are shown in Table 2 beneath. The RHM results reported in the published literature are also included in the table. ABC and DE are more viable than RHM. The results obtained by ABC are noticeably superior to those obtained by DE. The cost of generating is brought down to 1253.66 rupees per hour owing to the ABC solution for the OPF problem.

In addition, Performance evaluation of DE & ABC methods for the IEEE 11-bus system has been looked. In Fig. 2 only 24 iterations are needed by ABC to arrive at the best solution, but DE requires 56 iterations.

### Case 2 IEEE 30-bus systems

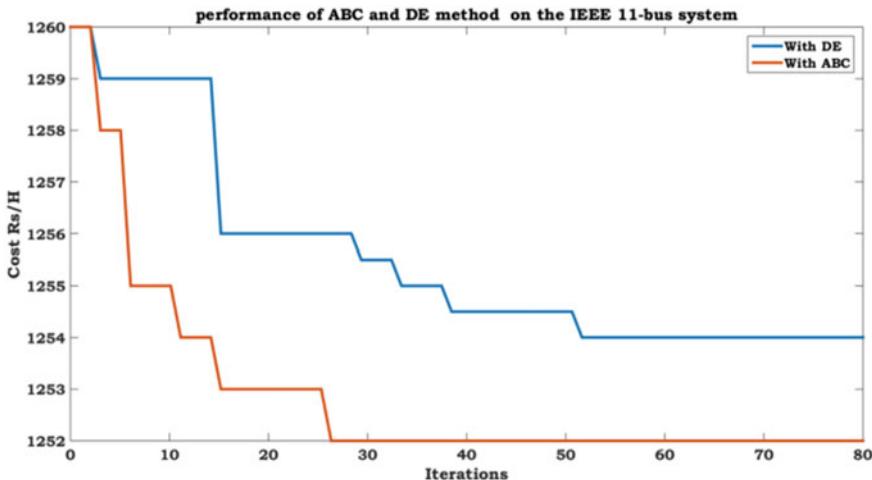
Figure 3 displays the algorithms' convergence plot in its entirety. Values in Table 4.

**Table 1** Data from the case 1

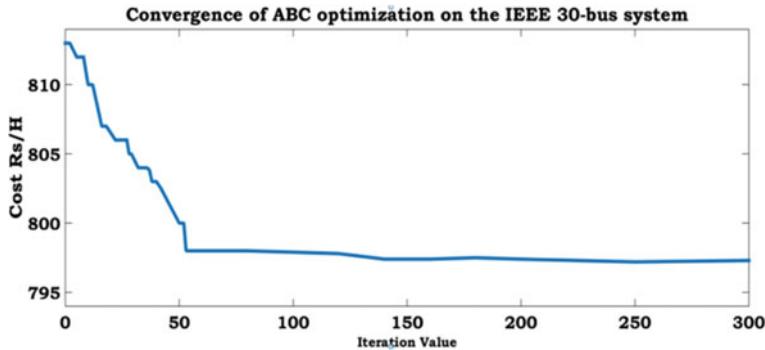
No.of generators	5
Transmission lines	17

**Table 2** Analyses of three different methods for case 1

	Parameter	ABC	DE	RHM
Active power (MW)	$P_1$	47.32	47.60	47.41
	$P_2$	74.67	74.62	74.71
	$P_3$	48.48	48.50	48.51
	$P_4$	47.54	47.55	47.61
	$P_5$	52.16	52.15	52.11
Qc (MVAR)	$Q_1$	26.10	26.10	26.30
	$Q_2$	43.60	43.80	43.70
	$Q_3$	0.00	0.04	0.10
	$Q_4$	14.48	14.49	14.20
	$Q_5$	13.44	13.44	13.40
Voltage angle (deg.)	$\delta_2$	1.39	1.39	1.39
	$\delta_2$	6.109	6.109	6.11
	$\delta_3$	2.79	2.79	2.80
	$\delta_4$	2.00	2.00	2.02
	$\delta_5$	1.64	1.64	1.64
TGC (Rs./h)	$C_{gen}$	1253.64	1254.52	1263.86

**Fig. 2** ABC and DE method performance assessment**Table 3** IEEE 30 bus-system data [20–22]

No.of generators	6 ( 1,2,5,8,11 &13 buses)
Tr. lines	41
Off-nominal tap ratios of transformers	4 (6–9, 6–10, 4–12, 28–27)
SVC places	10,12,15,17,20,21,23,24,29



**Fig. 3** Plot of the ABC method's convergence on cae-2

**Table 4** Analyses of three different methods for second case

	Parameter	ABC	DE [2]
Active power (MW)	$P_1$	176.476	176.259
	$P_2$	48.907	48.560
	$P_5$	22.222	21.340
	$P_8$	21.125	22.055
	$P_{11}$	12.307	11.778
	$P_{13}$	12.000	12.021
Bus voltage (pu)	$V_1$	1.100	1.099
	$V_2$	1.090	1.089
	$V_5$	1.096	1.065
	$V_8$	1.070	1.069
	$V_{11}$	1.100	1.096
	$V_{13}$	1.098	1.099
$R_c$ (MVAR)	$R^{12}$	4.978	4.556
	$R^{15}$	4.999	4.42
	$R^{15}$	4.290	4.17
	$R^{17}$	4.730	2.52
	$R^{20}$	4.999	2.091
	$R^{21}$	4.949	4.199
	$R^{23}$	4.418	2.55
	$R^{24}$	4.999	4.381
	$R^{29}$	2.090	2.750
TGC (Rs/h)	$TCgn$	799.265	799.29

The ongoing power flow approach, which involved calculating the MLP through the use of the ABC-based optimal power flow algorithm, was also applied to the system.

## 5 Conclusions

This paper examines the ABC metaheuristic approach to OPF control of a power system using SVC devices in great detail in an attempt to enhance voltage stability while also reducing generator fuel cost. Maximum loading point was discovered through CPF analysis and the recommended ABC algorithm was validated on IEEE 11-bus and 30-bus systems. Here, the ideal locations for the SVCs were determined with the aid of a sensitivity analysis. Lastly, ABC may be used to successfully handle the complicated nonlinear difficulties of power systems by virtue of its high qualities and quick convergence in a quick run time.

## References

1. Hingorani NG (1988) High power electronics and flexible AC transmission system. *IEEE Power Eng Rev* 8:3–4
2. Rong-Mow J, Nanming C (1995) Application of the fast Newton-Raphson economic dispatch and reactive power/voltage dispatch by sensitivity factors to optimal power flow. *IEEE Trans Energy Convers* 10:293–301
3. Karaboga D, Basturk B (2008) On the performance of artificial bee colony (ABC) algorithm. *Appl Soft Comput* 8:687–697
4. Liu J, Cao YC, Tang YY (1997) An evolutionary autonomous agents approach to image feature extraction. *IEEE Trans Evol Comput* 1:141–158
5. Abou El Ela AA, Abido MA, Spea SR (2010) Optimal power flow using differential evolution algorithm. *Electr Power Syst Res*
6. Abido MA (2002) Optimal power flow using particle swarm optimization. *Electr Power Eng Syst* 24(7)
7. Abou El Ela AA, Abido MA (2011) Differential evolution algorithm for optimal reactive power dispatch. *Electr Power Syst Res* 81:458–464
8. Abido MA, Bakhshwain JM (2005) Optimal VAR dispatch using a multi objective evolutionary algorithm. *Electr Power Eng Syst* 27
9. Kalpana N, VenuGopalaRao M (2020) Optimal location of UPFC to minimize the real power losses using Firefly Algorithm. *Int J Eng Adv Tech*
10. Ozturk A, Erdogmus P, Cobanli S, Tosun S (2010) Reactive power optimization with ABC algorithm. *Sci Res Essay* 5(19):2852
11. Linear programming, environment and electric engineering conference, 8–11 May 2011, Zagreb, Croatia: IEEE, pp 1–4
12. Basaran U (2004) Various power flow and economic dispatch analyses on 380 kV interconnected power system in Turkey. MSc, Anadolu University, Eskisehir, Turkey
13. Kalpana N, Rao MVG (2022) A standardized approach for evaluating UPFC's optimal location using metaheuristic algorithms to improve power quality. In: ICEARS 2022
14. Storn R, Price K (1996) Minimizing the real functions of the ICEC'96 contest by differential evolution. In: Proceedings of IEEE international conference on evolution computation, 20–22 May 1996, Munich, IEEE Germany, pp 842–884

15. Srivastava L, Prajapati BS (2012) Multi-objective reactive power optimization using artificial bee colony algorithm. IJEIT 2(1)
16. Shoultz RR, Sun DT (1982) Optimal power flow based upon p-q decomposition. IEEE Trans Power Appl Sys 101:397–405
17. Kalpana N, Rao MVG (2021) A generalized technique to assess the optimum location and performance analysis of UPFC using swarm-based algorithms. J Green Eng
18. El-Dib AA, Youssef H, El-Metwally MM, Osman Z (2005) Maximum loadability of power systems using hybrid particle swarm optimization. IEEE Trans Power Syst 20:1070–1078
19. Akay B, Karaboga D (2009) Parameter tuning for the artificial bee colony algorithm. In: International conference on computational collective intelligence technological and applications, 5–7 Oct 2009, Wroclaw, Springer, Poland, pp 608–619
20. Kalpana N (2023) Novel technique for evaluating optimal power flow and SVC performance using the ABC algorithm. In: 2023 Second international conference on electronis and renewable systems (ICEARS)
21. Kalpana N, Rao MVG (2019) Optimal location of UPFC to minimize the real power losses using NSPSO algorithm. IJTEE 8(12):924–931
22. Kalpana N, Rao MVG (2020) Impact of crow search algorithm to minimize transmission system power losses. JGE

# ELPDI: A Novel Ensemble Learning Approach for Pulmonary Disease Identification



Satwik Kulkarni, Sitanshu Hallad, Tanvi Bhujannavar, Abhishek S. Masur, Shankru Guggari, Uday Kulkarni, and S. M. Meena

**Abstract** Artificial intelligence has been a revolutionary concept for the healthcare sector in recent years. Deep Neural Networks (DNNs) are subdomains of machine learning which is a vital tool for applications such as diagnostic and therapy suggestions. Pulmonary diseases significantly influence the overall well-being of numerous individuals worldwide, greatly hampering their ability to lead a healthy and balanced life. The present study uses an ensemble technique to detect Pulmonary Diseases. Here, lung sounds obtained by auscultation are transformed into spectrograms and classified using Convolutional Neural Networks (CNN) trained on various architectures. The proposed study shows an accuracy of 97.3%.

**Keywords** Spectrogram · Ensembled learning · Deep neural network

## 1 Introduction

In today's world, Deep learning is an important technology to automate the system for real-world applications. As per World Health Organization (WHO) [1], respiratory diseases lead to more than three million deaths worldwide, making it the third leading cause of death. These diseases affect people's ability to lead a healthy life, hence the research for early detection and diagnosis of respiratory diseases is of great importance. Auscultation [2] is the oldest and one of the most effective methods of diagnosing lung diseases. It is an approach of listening to the body's internal sound for the purpose of examining the cardiac, respiratory systems, and alimentary canal to a certain extent. Auscultation is done using a stethoscope, and it is a medical device that has a resonator shaped like a disk at one end connected to one or two tubes which are in turn connected to earpieces. However, a traditional stethoscope is not capable of filtering external noise and the audio frequencies from the body, it is also not

---

S. Kulkarni (✉) · S. Hallad · T. Bhujannavar · A. S. Masur · S. Guggari · U. Kulkarni · S. M. Meena

KLE Technological University, Hubli, Karnataka, India  
e-mail: [kulkarnisatwik5@gmail.com](mailto:kulkarnisatwik5@gmail.com)

possible to record the sounds from the stethoscope to track the course of the disease. The diagnosis of these diseases using a stethoscope needs to be done by a highly trained and experienced medical professional. These limitations and conditions of the traditional stethoscope and the lack of availability of experienced medical professionals slow the diagnosis of diseases, hence providing a need for an electronic stethoscope [3] that uses machine learning [4] to detect the possibility of pulmonary diseases to assist the doctors in diagnosing patients thereby making the detection of such diseases fast and feasible. The lung sounds procured as a result of auscultation of the lungs can be generally classified into normal and abnormal sounds, and normal sounds are the lung sounds of a healthy patient and abnormal sounds contain additional sounds along with normal lung sounds. Machine learning algorithms classify these sounds to detect potential diseases a patient could be diagnosed with. One of the methods is the audio classification where the audio files of the dataset are classified using algorithms such as K Nearest Neighbor (k-NN) [5], Support Vector Machine (SVM) [6], or Multilayer Perceptron (MLP) [7], these classifiers provide accuracy in the range of 50 to 75%. In a different method, the audio files were converted into spectrograms [8] and the resulting images were classified using Deep Neural Networks (DNN) [9] which resulted in much greater accuracy. Although these methods are more accurate. In order to minimize prediction variance and error, the proposed methodology as represented in Fig. 4, employs ensemble learning [10] to integrate predictions from different neural networks.

In our study, Deep learning architectures, namely, VGG-16, ResNet-50, and Inception-V4 are used in pulmonary disease detection. Ensembling of these deep learning models is utilized for showcasing the performance improvement. The paper proceeds in the order of as follows.

Related works are described in Sect. 2. The methodology is discussed in Sect. 3. Results are illustrated in Sect. 4. Finally, conclusions are explained in Sect. 5.

## 2 Related Works

Lung sounds were classified as wheezes, crackles, and normal by Hai Chen et al. [11]. They proposed this novel method using the Optimized S-transform (OST). Murat et al. [7] in their work showed comparison between k-nearest neighbor (k-NN), support vector machines (SVM), and Gaussian Bayes (GB) algorithms in order to classify lung ailments using text and audio data. In the above works, accuracy ranged from 45 to 96%. Rupesh et al. [12] classified pulmonary diseases such as COPD and asthma they used FFT [13], STFT [14], spectrograms, and wavelet transform. They obtained an accuracy of 95%. Bardou Dalal et al. [15] used CNN and achieved an accuracy of 97%. However, this was highly dependent on the batch size, and the number of epochs, and was computationally expensive. Zeenat Tariq et al. [16] used spectrogram and CNN to obtain an accuracy of 97% tackling the issue of computational power.

A novel system called VGG Data STN with CNN (VDSNet) is introduced by Subrato Bharati et al. [17]. This hybrid approach combines CNN with VGG, data aug-

mentation, and spatial transformer network (STN). The researchers applied VDSNet to a dataset of NIH chest X-ray images obtained from the Kaggle repository. The validation accuracy of VDSNet for the complete dataset is reported as 73%. In comparison, vanilla grey, vanilla RGB, hybrid CNN, VGG, and modified capsule network achieved accuracy scores of 67.8%, 69%, 69.5%, and 63.8% respectively. Apart from achieving good accuracy with less computational power, there was still a need to reduce the uncertainty and variability in the prediction of the CNN models.

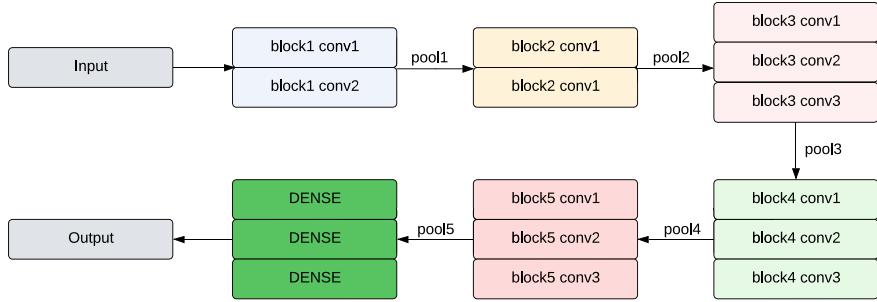
Based on the literature, most of the time DNN models provide higher classification accuracy as compared to other state-of-art methods such as SVM and k-NN. Hence the present study uses VGG-16, Inception-V4 and ResNet-50, which are discussed as follows.

## 2.1 *VGG-16*

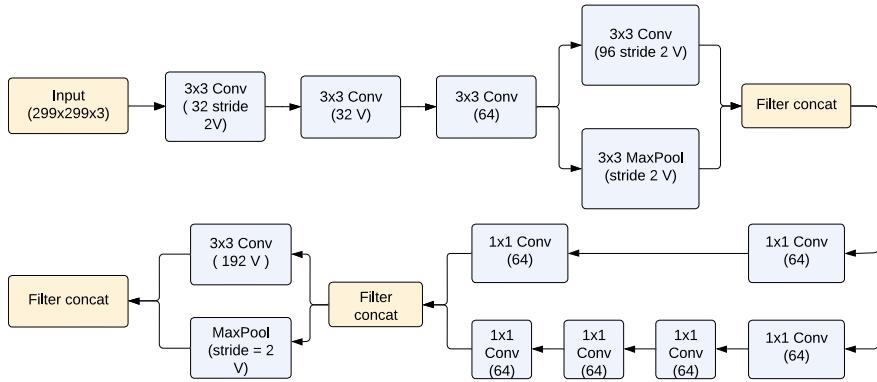
The vgg-16 [18, 19] architecture is mostly employed in picture categorization. As seen in Fig. 1, it has 16 parametric layers, 13 of which are CNN layers, 5 pooling layers, and 3 Dense layers, for a total of 21 levels. The input image size in the VGG-16 architecture with three RGB channels is (224,244) [18]. This approach has the advantage of focusing on various levels of the architecture such as Convolution layers and employing the same max pool and padding layer instead of storing all of the hyper-parameters [19, 20]. The convolutional layers are used to identify features from the input image, while the Fully Connected (FC) layers are used for classification. The convolutional layers are arranged in five blocks, each consisting of 2–3 convolutional layers. Subsequently, a maximum pooling layer is applied. The first block employs a sole convolutional layer with 64 filters, and this is succeeded by a max pooling layer. The second and third blocks are composed of a pair of convolutional layers each, with 128 and 256 filters respectively, and are succeeded by a max pooling layer. The fourth block encompasses three convolutional layers with 512 filters, followed by a max pooling layer. The fifth block contains three convolutional layers with 512 filters, but no max pooling layer. After the convolutional layers, the network has three FC layers. The initial fully connected (FC) layer consists of 4096 units, followed by another FC layer with 4096 units, and finally, a third FC layer with 1000 units, representing each class in the ImageNet dataset. The output from the third FC layer is passed through a softmax function, generating a probability distribution across the 1000 classes. The architecture of VGG-16 has been widely used as a starting point for many other image recognition models [18].

## 2.2 *Inception-V4*

Inception-V4 [21, 22] is a DNN model with 27 layers of depth which consists of 2 convolution layers, 4 max pool layers, 9 inception layers, 1 avg pool, 1 dropout,



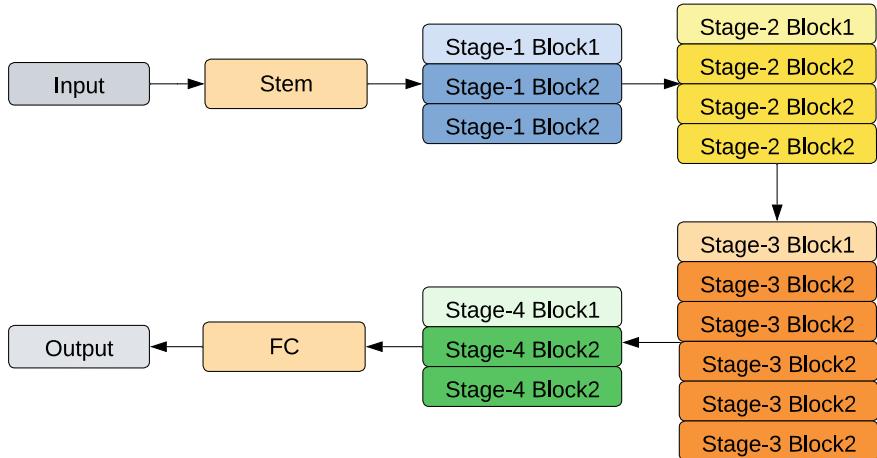
**Fig. 1** VGG-16 architecture



**Fig. 2** Inception-V4 architecture

1 linear, and 1 softmax layer altogether, as you can see in the Fig. 2. This architecture makes use of the sparsely connected networks that help increase the depth and width of the networks, at the same time, reducing the over-fitting and saving on the computational budget.

The key notion behind employing inceptions is that we have many operations at one layer that are all launched at the same time. They collaborate to produce an ensemble of output characteristics, resulting in a model that is adaptable to any incoming input. Batch normalization is used in auxiliary classifiers, the RMSProp optimizer, factorized 7\*7 convolutions, and label smoothing in the Inception-V4 [22, 23].



**Fig. 3** ResNet architecture

### 2.3 ResNet-50

ResNet-50 [24] is a variation of the ResNet model with 48 Convolution layers, 1 Average Pool, and 1 MaxPool layer as seen in Fig. 3. It is capable of performing  $3.8 \times 10^9$  32-bit precision(float) calculations. It is a well-known ResNet model.

The suggested ResNet [24, 25] was designed to address the degradation issue in deep networks. Previously, there was a fundamental issue with deep networks in that as the depth of the networks increased, the accuracy became saturated and rapidly degraded. To address this issue, the authors [24] presented the hypothesis together with actual proof of skip connections, which greatly reduced the degradation problem. These were referred to as residual connections. This allowed them to create one of the deepest networks without sacrificing the model's accuracy. Then the model is trained using ResNet-50 architecture where feature extraction is done deeply which results in more accuracy.

Various ensemble techniques are present in the literature with respect to deep learning models [26–29]. The present study utilizes popular deep learning techniques for pulmonary disease identification which uses images of spectrogram from audio files of lunges collected via auscultation. In the next section, present study is explained in detail.

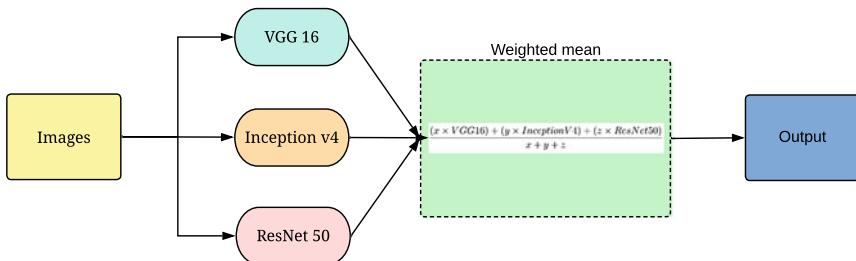
### 3 Ensemble Learning for Identification of Pulmonary Disease

In this section, the Ensemble technique for pulmonary disease identification (ELPDI) is described. The present study uses the combination of three different Deep Learning Models trained on standard Deep Learning architectures, namely, ResNet-50 [25], Inception-V4 [22], and VGG-16 [18]. Initially, the audio files are converted to spectrograms. Spectrograms are visual representations of audio signals. They can be useful for a variety of tasks, such as speech recognition and audio analysis. Because a spectrogram shows the frequencies present in an audio signal over time, it can provide more detailed information about the audio than the raw waveform of the signal. The ability to convert audio signals into spectrograms has made it possible to use deep learning methods for a variety of audio analysis tasks as shown in Fig. 4.

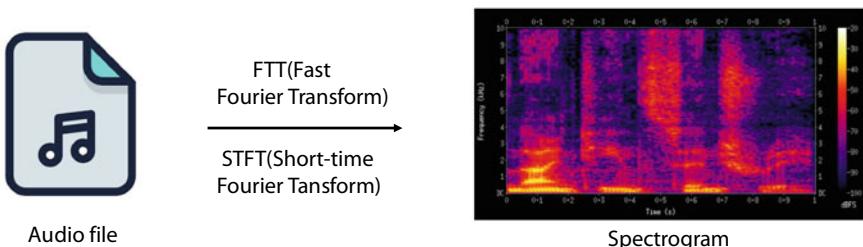
#### 3.1 Converting Audio Files into Spectrograms

In ELPDI, the audio files of lung sounds are recorded with the help of an electronic stethoscope and are converted to respective spectrograms as shown in Fig. 5.

A spectrogram [8] is analogous to a signal snapshot or image. It represents time along the vertical axis and frequencies along the horizontal axis. A spectrogram also



**Fig. 4** Audio classification using ELPDI with DNN models



**Fig. 5** Representation of conversion of audio file to spectrogram

displays signal intensity using colors; the brighter the hue, the greater the signal's energy. It primarily describes how the signal intensity is dispersed across all frequencies in the transmission. To obtain the frequencies contained in the signal, it employs the Fast Fourier Transform (FFT) [30] and Short-Time Fourier Transform (STFT) [31]. Using the FFT to create a spectrogram is a digital technique. Digitally sampled data is split into chunks in the time domain, and the Fourier transformation is employed to determine the intensity of the frequency spectrum for each chunk. Every individual chunk then correlates to a vertical line in an image, representing a magnitude versus frequency measurement at a certain instant in time. These spectrums are then arranged side by side. This procedure computes the square of the magnitude obtained from the Short-Time Fourier Transform (STFT) as shown in Eq. 1. For the signal  $s(t)$  and for a window width ( $\omega$ )

$$\omega, \text{spectrogram}(t, \omega) = |\text{STFT}(t, \omega)|^2 \quad (1)$$

FFTs divide a signal into separate spectral components and offer frequency information about the signal as a result. STFT is employed to calculate sinusoidal frequency and phase attributes of small signal segments as they progress with time. The generated spectrogram is further used for the classification of various diseases using Deep Neural Networks.

### 3.2 Classification of Generated Spectrograms

The results were promising after developing the model using DNN architectures. However, these neural networks exhibited high variance, which can pose issues when developing a final model for prediction, therefore we were looking for a method to enhance accuracy while decreasing loss and variation in outcomes when trained using diverse datasets. Here we propose a solution to this problem using Ensemble learning for pulmonary disease identification (ELPDI). Ensemble learning is the systematic generation and combination of multiple models to address a given computational intelligence problem [10]. This method is mainly used to boost the model's performance with minimal loss in accuracy. By using this method the models or classifiers of various architectures (VGG-16, Inception-V4, ResNet-50) which are discussed above are systematically integrated and predictions are made by taking the mean of probabilities of the output of all 3 classifiers which yielded minimal prediction variability and maximum accuracy. ELPDI is represented in Fig. 3.

$$\frac{(x \times \text{VGG16}) + (y \times \text{InceptionV4}) + (z \times \text{ResNet50})}{x + y + z} \quad (2)$$

Initially, the image (spectrogram) dataset is trained using VGG-16, Inception-v4, and ResNet-50. The ensemble approach is then used to produce the output by taking the weighted mean of the distribution of probabilities of an individual model with

the help of the equation, which happens to be the output when the softmax activation function is applied [32]. This output is used for prediction. Values of  $x$ ,  $y$ , and  $z$  used in the Eq. 2 decide the priority given to a particular architecture.

## 4 Results

This section focuses on analyzing the performance of ELPDI using the ICBHI 2017 challenge database taken from [33] and compares the results of ELPDI and models built with ResNet-50, VGG-16, and Inception-V4 architectures.

### 4.1 Dataset Description

The Dataset [33] contains respiratory sounds, which are critical markers of respiratory health and diseases. The sound produced while a person is breathing is closely tied to the movement of air and alterations in lung tissue. Digital stethoscopes can be used to capture these sounds. This digital data opens the door to applying machine learning algorithms to detect respiratory ailments at an early stage.

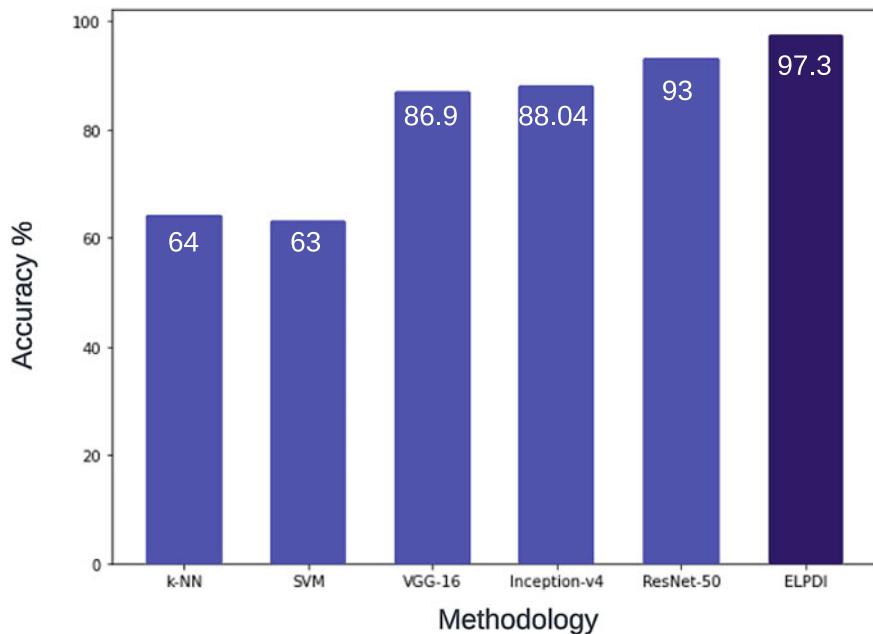
This dataset contains 920 recordings from 126 patients, spanning a range of lengths between 10 and 90s. There are 330 min of recordings totaling 6898 respiratory cycles, including 1864 with crackles, 886 with wheezes, and 506 with both wheezes and crackles. The dataset comprises a combination of clear sounds and recordings with noise that imitate real-world situations. The age range of the patients includes children, adults, and the elderly. Diseases and the total count of samples present in the dataset are represented in Table 1.

Figure 6 shows the accuracy of the various methods which were used to classify lung diseases. The proposed model approximately shows 10.5%, 9.2%, and 4.3% improvement in classification accuracy as compared to individual models built on VGG-16, Inception-V4, and ResNet-50 architectures respectively (Table 2).

Table 3 and Fig. 7 compare time taken by ELPDI and other CNN models to give a prediction.

**Table 1** Distribution of audio files for each pulmonary disease

Name of disease	Number of audio files
Asthma	1
Bronchiectasis	29
COPD	785
Healthy	35
LRTI	2
Pneumonia	37
URTI	31



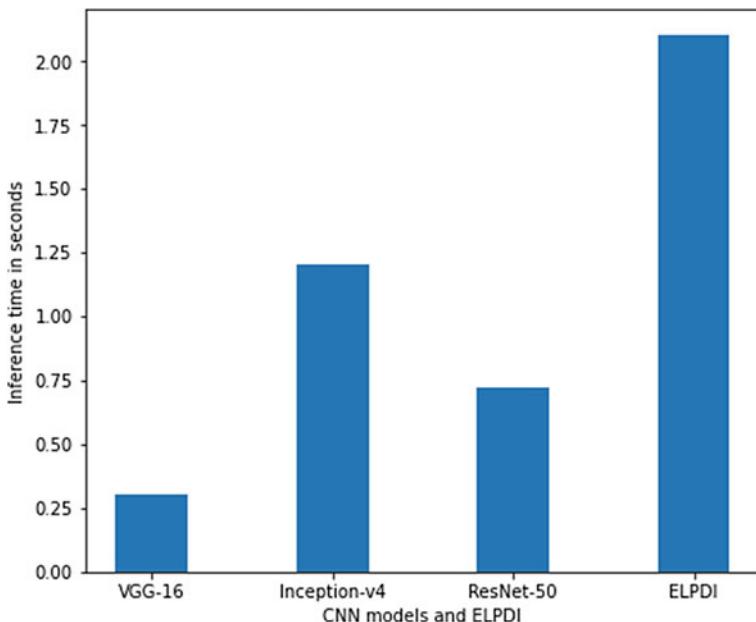
**Fig. 6** Comparing ELPDI with other state-of-art techniques

**Table 2** Accuracies of various methods to classify lung diseases

Method	Accuracy (%)
k-NN	64
SVM	63
VGG-16	86.9
Inception-v4	88.04
ResNet-50	93
<b>ELPDI</b>	<b>97.3</b>

**Table 3** Inference time of individual CNN models and ELPDI

Model	Inference time (in s)
VGG-16	0.3
Inception-v4	1.2
ResNet-50	0.72
<b>ELPDI</b>	<b>2.1</b>



**Fig. 7** Comparison of inference time

## 5 Conclusion

ELPDI is a method to identify respiratory diseases based on the respiratory sound dataset that contains audio files. These audio files are converted to spectrograms and analyzed by ensemble learning which is culminated in three different architectures, i.e., Inception-v4, ResNet-50, and VGG-16. The mean of the probability distribution of the three architectures is obtained for the final result of classification. This enables us to reach an accuracy of 97.3%. Anyone who wishes to work on this topic can use the image dataset available at this [url](#). This study can be further utilized for early diagnosis of pulmonary diseases. Future studies can be dedicated to creating an electronic stethoscope that can incorporate this model.

## References

1. Chronic obstructive pulmonary disease (COPD) (2022) [https://www.who.int/news-room/fact-sheets/detail/chronic-obstructive-pulmonary-disease-\(copd\)](https://www.who.int/news-room/fact-sheets/detail/chronic-obstructive-pulmonary-disease-(copd)). Accessed 05 Dec 2022
2. Auscultation—an overview|ScienceDirect Topics. <https://www.sciencedirect.com/topics/nursing-and-health-professions/auscultation>. Accessed 09 Dec 2022
3. Fraiwan M, Fraiwan L, Khassawneh B, Ibnian A (2021) A dataset of lung sounds recorded from the chest wall using an electronic stethoscope. Data Brief 35:106913

4. Yu KH, Beam AL, Kohane IS (2018) Artificial intelligence in healthcare. *Nat Biomed Eng* 2(10):719–731
5. Harrison O (2019) Machine learning basics with the K-nearest neighbors algorithm. <https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>. Accessed 09 Dec 2022
6. Palaniappan R, Sundaraj K, Ahamed NU (2013) Machine learning in lung sound analysis: a systematic review. *Biocybern Biomed Eng* 33(3):129–135
7. Aykanat M, Kılıç Ö, Bahar K, Saryal SB (2020) Lung disease classification using machine learning algorithms. *Int J Appl Math Electr Comput* 8(4):125–132
8. Spectrogram—an overview!ScienceDirect Topics. <https://www.sciencedirect.com/topics/engineering/spectrogram>. Accessed 11 Dec 2022
9. Sze V, Chen YH, Yang TJ, Emer JS (2017) Efficient processing of deep neural networks: a tutorial and survey. *Proc IEEE* 105(12):2295–2329
10. Cerliani M (2020) Neural networks ensemble. <https://towardsdatascience.com/neural-networks-ensemble-33f33bea7df3>. Accessed 09 Dec 2022
11. Chen H, Yuan X, Pei Z, Li M, Li J (2019) Triple-classification of respiratory sounds using optimized s-transform and deep residual networks. *IEEE Access* 7:32845–32852
12. Dubey R, Bodade RM (2019) A review of classification techniques based on neural networks for pulmonary obstructive diseases. *Proc Recent Adv Interdisc Trends Eng Appl (RAITEA)*
13. Nussbaumer HJ (1981) The fast fourier transform. In: *Fast fourier transform and convolution algorithms*, pp 80–111. Springer
14. Moorer JA (2017) A note on the implementation of audio processing by short-term fourier transform. In: *2017 IEEE workshop on applications of signal processing to audio and acoustics (WASPAA)*. IEEE, pp 156–159
15. Bardou D, Zhang K, Ahmad SM (2018) Lung sounds classification using convolutional neural networks. *Artif Intell Med* 88:58–69
16. Tariq Z, Shah SK, Lee Y (2019) Lung disease classification using deep convolutional neural network. In: *2019 IEEE international conference on bioinformatics and biomedicine (BIBM)*. IEEE, pp 732–735
17. Bharati S, Podder P, Mondal MRH (2020) Hybrid deep learning for detecting lung diseases from x-ray images. *Inf Med Unlocked* 20:100391
18. Vgg16 architecture (2019) <https://iq.opengenus.org/vgg16/>. Accessed 11 Dec 2022
19. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arxiv:1409.1556](https://arxiv.org/abs/1409.1556)
20. Team GL (2021) Introduction to VGG16!What is VGG16? <https://www.mygreatlearning.com/blog/introduction-to-vgg16/>. Accessed 11 Dec 2022
21. Szegedy C, Ioffe S, Vanhoucke V, Alemi A (2016) Inception-v4, inception-resnet and the impact of residual connections on learning. [arxiv: 1602.07261](https://arxiv.org/abs/1602.07261)
22. Tsang SH (2020) Review: inception-v4—evolved from GoogLeNet, merged with ResNet Idea (Image Classification). <https://towardsdatascience.com/review-inception-v4-evolved-from-googlenet-merged-with-resnet-idea-image-classification-5e8c339d18bc>. Accessed 11 Dec 2020
23. Brital A (2021) Inception V4 CNN architecture explained. <https://medium.com/@AnasBrital98/inception-v4-cnn-architecture-explained-23a7fe12c727>. Accessed 11 Dec 2022
24. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. [arxiv: abs/1512.03385](https://arxiv.org/abs/1512.03385)
25. Understanding ResNet50 architecture (2020) <https://iq.opengenus.org/resnet50-architecture/>. Accessed 11 Dec 2022
26. Altan G, Kutlu Y, Gökcen A (2020) Chronic obstructive pulmonary disease severity analysis using deep learning on multi-channel lung sounds. *Turk J Electr Eng Comput Sci* 28(5):2979–2996
27. Mo A, Gui E, Fletcher RR (2022) Use of voluntary cough sounds and deep learning for pulmonary disease screening in low-resource areas. In: *2022 IEEE global humanitarian technology conference (GHTC)*. IEEE, pp 242–249

28. Nathan V, Vatanparvar K, Chun KS, Kuang J (2022) Utilizing deep learning on limited mobile speech recordings for detection of obstructive pulmonary disease, pp 1338–1341
29. Srivastava A, Jain S, Miranda R, Patil S, Pandya S, Koticha K (2021) Deep learning based respiratory sound analysis for detection of chronic obstructive pulmonary disease. PeerJ Comput Sci 7:e369
30. Maklin C (2019) Fast fourier transform. <https://towardsdatascience.com/fast-fourier-transform-937926e591cb>. Accessed 20 Dec 2022
31. Short-Time Fourier Transform—an overview!ScienceDirect Topics. <https://www.sciencedirect.com/topics/engineering/short-time-fourier-transform>. Accessed 20 Dec 2022
32. Koech KE (2021) Softmax activation function—how it actually works. <https://towardsdatascience.com/softmax-activation-function-how-it-actually-works-d292d335bd78>. Accessed 9 Dec 2022
33. Rocha B, Filos D, Mendes L, Vogiatzis I, Perantoni E, Kaimakamis E, Natsiavas P, Oliveira A, Jácome C, Marques A et al (2017) A respiratory sound database for the development of automated classification. In: International conference on biomedical and health informatics. Springer, pp 33–37

# JAYA-Based Task Scheduling Algorithm in Fog-Cloud Environment



Nupur Jangu and Zahid Raza

**Abstract** In the new era of the Internet of Everything (IoE), the data generated is vast and varied and sending it to the cloud generates latency, congestion and increases delay, cost, and energy usage. As a result, fog and other related edge computing paradigms are being used as complementary to the cloud. These computing paradigms execute real-time requests with minimal latency, provide location awareness, resource sharing and mobility while preventing the cloud layer from overloading. However, resource management remains a bottleneck in these computing environments in spite of an improved user experience. This research presents an optimization approach I-JAYA for task scheduling in an integrated fog-cloud environment for resource management using the optimization algorithm JAYA with an improvement in its exploration phase. Simulation study reveals the outperformance of I-JAYA over its parent JAYA in terms of the average make-span time realized for jobs of various sizes.

**Keywords** Cloud computing · IoT · Fog computing · Meta-heuristic · Scheduling

## 1 Introduction

Cloud computing has been widely used as a computing platform for decades. But with the exponential growth of the number of IoT devices in recent years, it has started facing problems. Higher latency, bandwidth consumption, and high computation cost due to the massive volume and variety of data produced are a few of these encountered problems [1]. Cisco predicted the rise in Internet-connected devices to 50 billion by 2020, which is already crossed. With the need to mitigate these challenges and the huge data generated from IoT devices, a new computing paradigm, fog was coined by Cisco. It aimed to be an appropriate option for devices with limited resources and users that require a quick response within the strict deadline thus offering an improved Quality of Service (QoS). A hybrid environment of computing paradigms

---

N. Jangu (✉) · Z. Raza

School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India  
e-mail: [nupur68\\_scs@jnu.ac.in](mailto:nupur68_scs@jnu.ac.in)

is a viable model that enables the low latency serving and management of enormous amounts of data created by billions of IoT devices and applications thus moving towards an integrated fog-cloud architecture [2].

The allocation of user tasks to appropriate resources determines the system's effectiveness and impacts both users and suppliers. Incorrect decisions can jeopardize user access and fail to live up to their expectations if the application is delayed. Job scheduling for resource management in cloud, fog and edge seek to maximize the desired objectives, e.g., turnaround time, response time, throughput, and security to name a few. Based on these parameters, the resources are provisioned for the incoming job streams. Minimizing the task execution cost is any task scheduling algorithm's primary performance metric. The problem of task scheduling has been established as an NP-Complete [3] and optimization technique suggests a mean to solve it effectively. These techniques may consider various performance parameters such as completion time, cost, resource utilization, etc. The data generated from IoT devices is both voluminous and heterogeneous. Since, most of this data corresponds to real time situations and requires delay sensitive decision making, it demands for exceptionally low latency handling.

In the integrated fog-cloud environment, there are many other challenges such as improper task classification and inefficient management of resources. In addition, load uncertainty, increased delay, and lack of effective offloading decisions also affect the performance greatly. These are some of the major limitations of the existing approaches, which served as a motivation for this research. This work aims to address the challenge of distributing the edge device-generated IoT requests to the appropriate resources in the integrated fog-cloud environment for the following objectives:

- To achieve an improved QoS in terms of the completion of tasks by classifying data based on priority and deadline.
- To minimize the execution time of the tasks by scheduling the tasks based on multiple aspects using the JAYA [4] optimization algorithm with improvement in the exploratory phase.
- To maximize the resource utilization of the tasks by adequately managing resources in the fog and cloud tier.

## 2 Related Work

Many researchers have worked successfully in resource provisioning, particularly targeting task scheduling in distributed environments like cloud, fog, and edge from a long time. The task scheduling problem falls into the category of NP-Hard and therefore metaheuristics have been actively reported in the literature for efficient resource provisioning. As an example, in [2, 5–7], authors have employed metaheuristics approaches like Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Cuckoo Search Algorithm (CSA), and such similar methods.

Task scheduling is governed by many factors from both the user and provider perspectives. However, the main objective remains as the minimization of the execution time of the user tasks. The state of the art has many exemplary works both in the field of cloud-only and hybrid architecture of fog-cloud addressing these issues. Few such works, e.g., [8–11], target minimization of execution of tasks to improve the overall QoS. The practicability and competence of the JAYA algorithm make it applicable in solving diverse real-world applications [12]. These applications include mathematical functions [13, 14], feature selection [15], image processing [16], energy [17], communication [18], planning and scheduling [19], range-based localization [20] among many other possibilities.

## 2.1 JAYA

JAYA is a single-phase algorithm-specific parameter less algorithm and is easier to apply than other meta-heuristics available. It was developed while considering the requirement of tuning the algorithm-specific parameters and their effect on the increase of computational effort and yield of optimal local solution [4]. JAYA runs in a single phase and can solve both limited and unconstrained optimization problems. Because of this, it delivers the optimized output after a small number of function evaluations.

Authors in [18] proposed a method for balancing load using the JAYA approach for cloud computing as it uses fewer control parameters and is a straightforward method. They tried to map tasks in a specific way to balance the load equitably by considering various virtual machines and the time required for all tasks to complete. The algorithm's effectiveness is assessed by comparing its experimented results using a variety of metrics with PSO and GA. The simulation results demonstrate that the proposed JAYA algorithm outperforms the GA and PSO by decreasing the reaction time.

In another work, [21], workflows are scheduled using the JAYA algorithm in a cloud computing environment to reduce the execution cost, make-span, and convergence ratio of a task resource mapping problem. Compared to other algorithms, JAYA produces an optimal mapping in the shortest time. Although ACO and honeybee algorithms provide optimal solutions more slowly, they have lower execution costs. It was found that the proposed method is quick since it has no parameters, which helps it in overcoming the drawback of previous algorithms that have adjustable parameters.

For a balanced system and appropriate job mapping into virtual machines, the work [22] proposed a binary JAYA-based scheduling method combined with the load balancing mechanism. Additionally, the binary load-scheduling JAYA incorporates a load-balancing strategy to handle the degree of imbalance among virtual machines. According to the findings of two test scenarios with dynamically independent jobs and heterogeneous data, it is statistically superior to others in resource consumption and make-span.

In [23], an enhanced version of the JAYA optimization approach named IJAYA for task scheduling on heterogeneous resources that draws inspiration from nature is proposed. The continuous numbers are changed into discrete values using a binary IJAYA, and the algorithm's efficacy is assessed using a real scenario dataset against the standard JAYA, Bird Swarm Optimization (BSO), Particle Swarm Optimization (PSO), and Salp Swarm Algorithm (SSA). Compared to other examined algorithms, the results show considerable improvements in make-span, degree of imbalance (DOI), response time, and resource consumption.

Similarly, in [24], the authors proposed a binary self-adaptive JAYA-based load scheduling algorithm. The suggested algorithm was experimented with in a heterogeneous environment and outperformed BSO, modified particle swarm optimization (MPSO), and the standard JAYA. For evaluation, the authors used a dataset logged by NASA for the various QoS scheduling and logged considerable improvements in terms of make-span, average resource utilization, and load-balancing.

### 3 Task Scheduling Using I-JAYA

This study proposes a modified version of the standard JAYA, i.e., I-JAYA as the foundation for the suggested job scheduling algorithm in the hybrid environment. The classification of data in an integrated environment of fog-cloud has its vital significance. This work incorporates a classification phase for the incoming tasks in which these tasks are classified based on priority and deadline into four categories: highly intensive, intensive, moderate, and low intensive to know the best execution location. The workflow of the proposed scheduler has been depicted in Fig. 1.

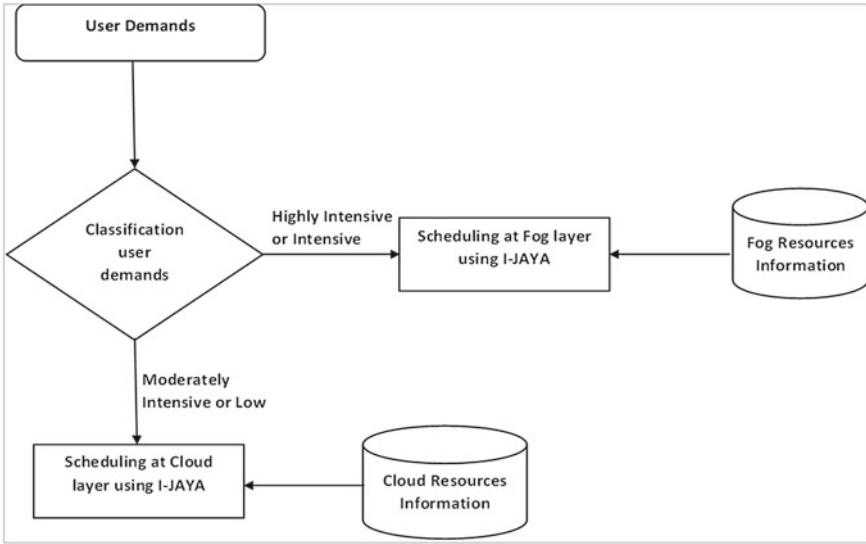
#### 3.1 I-JAYA Algorithm

For some optimization issues, JAYA can easily get stuck in local space. Therefore, its exploitation and exploration capabilities need to be controlled and tuned for improving its functionality and applicability. The proposed Improved-JAYA (I-JAYA) solves the scheduling issue in a hybrid fog-cloud scenario and enhances the finish time for all users' tasks on the available VMs. The fitness function  $F_t$  can be computed as,

$$F_t = \max\{ECT_{ij}\} \forall i \in [1, k] \text{ mapped to } j^{\text{th}} \text{ VM}, j = 1, 2, \dots, n \quad (1)$$

where  $\max\{ECT_{ij}\}$  denotes a maximum time to complete Task  $i$  on  $Vm_j$ .

I-JAYA improves the exploration phase to achieve faster convergence and reduce the search time. Three solutions were chosen randomly from the population, and each is updated, as necessary. This can be formulated as



**Fig. 1** Scheduler workflow

$$\zeta_{ij}'(t+1) = \zeta_{ij} + rn \times (\zeta_{rn_1}(t) - \zeta_{rn_2}(t)) - (1 - rn)(BestSol_j - \zeta_{rn_3}(t)) \quad (2)$$

The pseudocode of the I-JAYA approach is provided in Fig. 2.

## 4 Performance Evaluation

This section presents the experimental analysis of I-JAYA on the fog-cloud environment. The scheduler optimizes the execution IoT tasks scheduling based on priority, deadline, and resource needs. Accordingly, the requests are categorized according to priority and deadline, and then they are scheduled either on the fog or the cloud.

### 4.1 Simulation Study

A hybrid fog-cloud environment with heterogeneous task transfer was simulated for the trials in MATLAB that replicates a real-world cloud environment. The simulation characteristics adopted in the work are outlined in Table 1. The MATLAB toolbox is utilized for efficient simulation. Three thousand iterations were performed during all 20 rounds of the experiments.

The goal of simulation for I-JAYA is to determine the make-span of the two meta-heuristic scheduling techniques, viz, JAYA and I-JAYA under consideration.

---

**Pseudocode**

I-JAYA-based task scheduling ()

**Input:**  $T_i^C$  and  $T_i^R$

**Output:**  $\zeta^*$

**Begin**

- Perform initialisation
- Evaluate  $\zeta$  using  $F_t$  fitness function and select one in  $\zeta^*$
- Sort the population: (BestSol and WorstSol are the best and worst solutions, respectively)
- Set  $t = 1$
- While**  $t < t_{max}$
- For**  $j = 1:N$
- Set  $r \in [0,1]$
- Evaluation of  $\zeta_{ij}$  and reuse it if found better
- Perform updation of  $\zeta$
- $t = t + 1$
- End for**
- For**  $j = 1:N$
- Generate  $r$  in the range of 0 to 1
- Perform updation of  $\zeta_{ij}$  using (2)
- End for**
- End while**

**End**

---

**Fig. 2** Pseudocode of I-JAYA

**Table 1** Simulation attributes

Entity	Parameter	Values
User requests	Number of requests	600–3000
	Length	10–50 (Mb)
	Priority	High/low
	Deadline	High/low
Virtual machine (Cloud)	Number of virtual machines	30% of the total on the cloud
	Processing capacity range	[100, 1000] (MIPS)
Virtual machine (Fog)	Number of virtual machines	70% of the total on the cloud
	Processing capacity range	[10, 100] (MIPS)

Different test scenarios are utilized to determine how well these two operate based on the ECT to guarantee a quick turnaround time and effective resource use. The task classification, done before the scheduling phase in the suggested model, enhances total performance by categorizing them based on priority and deadline. These two factors make it possible and improve the overall execution time of tasks.

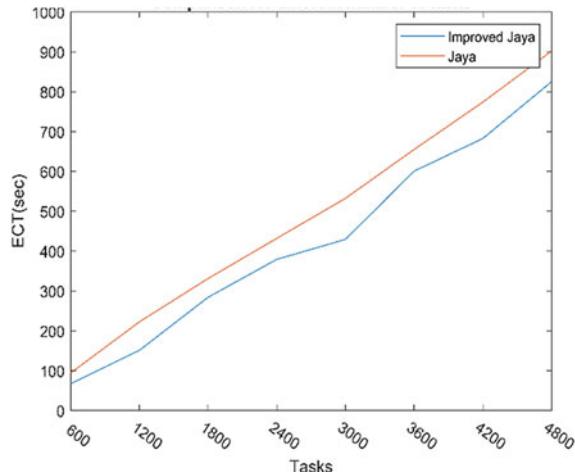
The efficiency of the suggested I-JAYA algorithm and the conventional JAYA algorithm was assessed by submitting a range of tasks, from 600 to 4800, to determine an overall trend of the two algorithms' performance as presented in Fig. 3. This is crucial because a diverse range of data sizes could support the extensive evaluation of task

transfer scheduling. The scheduling gets increasingly challenging as the data amount increases. However, I-JAYA outperforms the original JAYA method and reports even better results as the batch size increases. The discrepancy in ECT reported by both systems grows with an increase in the batch size. The broader disparity between the two led to the conclusion that I-JAYA is better suited for situations involving huge datasets, which is quite appropriate, especially when considering IoT devices as the task generators in real-world situations.

The varying range of tasks was also scheduled in five different batches using JAYA and I-JAYA and various trials were taken to justify the effectiveness of the algorithms. Table 2 summarizes the optimized ECT based on the various experiment scenarios.

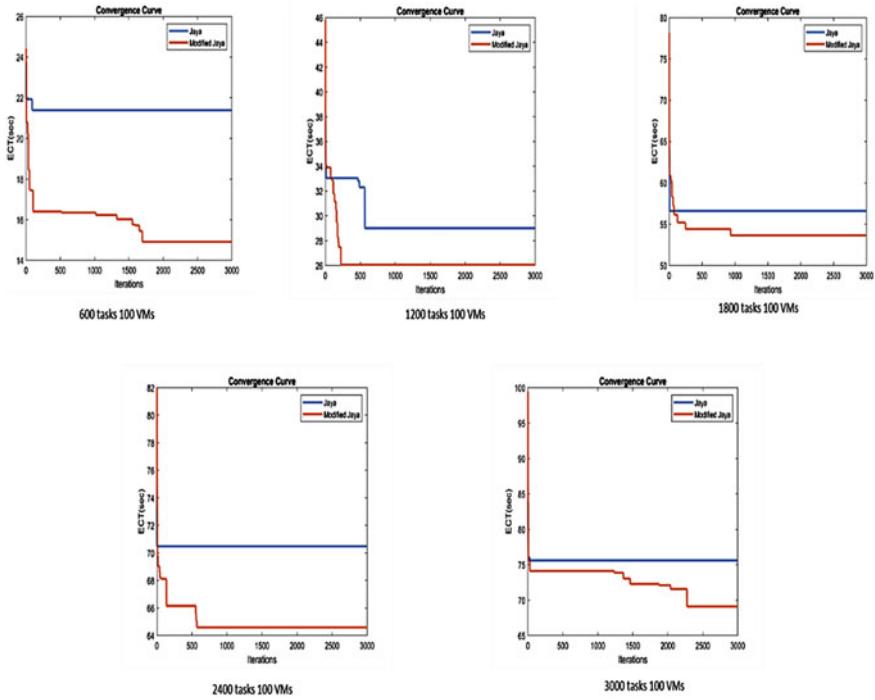
Figure 4a–e shows the Expected Completion Time (ECT) for all batches from 600 to 3000 tasks based on both the used algorithms through convergence graphs. It is worth mentioning that the I-JAYA converges faster for smaller to bigger batch sizes because I-JAYA has a better exploration capability than conventional JAYA, it can

**Fig. 3** Comparison of ECTs of different batches of tasks



**Table 2** Summary of ECT experimental results

Summary of ECT					
S. no	Tasks	JAYA algorithm		Improved JAYA algorithm	
		Best	Worst	Best	Worst
1	600	21.383	24.4074	<b>14.9124</b>	24.4074
2	1200	28.9985	45.8309	<b>26.4866</b>	45.8309
3	1800	56.6058	78.1034	<b>53.6445</b>	78.1034
4	2400	70.4798	81.9732	<b>64.5875</b>	81.9732
5	3000	75.6035	99.4457	<b>66.7704</b>	99.4457



**Fig. 4 a–e** Convergence graphs

achieve faster convergence by cutting down on searching time. In I-JAYA, this was made possible by exploring areas other members had not yet been able to investigate, selecting any three answers randomly, and updating the population, as necessary.

## 5 Conclusion and Future Work

In computing models like Cloud, Fog, and Edge, resource management using task scheduling is of utmost importance to minimize the make-span. This work proposed an I-JAYA-based scheduling model to map user tasks onto resources targeting minimum completion time of real-time tasks generated through IoT devices in a fog-cloud integrated architecture. The model achieves faster convergence owing to fewer algorithm-specific parameters and with an improvement in the exploratory phase thus reduces the search time. It is a superior convergent algorithm since it combines the best and worst solutions to arrive at the optimum mapping. Consequently, I-JAYA can be effectively employed for job scheduling in cloud computing environment. Under various test scenarios, I-JAYA-based scheduler surpasses the conventional JAYA, demonstrating its efficacy.

This work can be extended by designing a multi-objective model that reduces the reaction time and energy consumption using the integrated fog-cloud architecture's thereby improving QoS for the users while lowering the service provider's cost.

## References

1. Raza Z, Jangu N (2022) Workload classification. *Int J Syst Serv-Orient Eng* 12:1–14. <https://doi.org/10.4018/ijssoe.297135>
2. Aburukba RO, AliKarrar M, Landolsi T, El-Fakih K (2020) Scheduling Internet of Things requests to minimize latency in hybrid Fog–Cloud computing. *Futur Gener Comput Syst* 111:539–551. <https://doi.org/10.1016/j.future.2019.09.039>
3. Kalra M, Singh S (2015) A review of metaheuristic scheduling techniques in cloud computing. *Egyptian Inform J* 16:275–295. <https://doi.org/10.1016/J.EIJ.2015.07.001>
4. Venkata Rao R (2016) Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *Int J Ind Eng Comput* 7:19–34. <https://doi.org/10.5267/j.ijiec.2015.8.004>
5. Raju R, Babukarthik RG, Chandramohan D, et al (2013) Minimizing the makespan using Hybrid algorithm for cloud computing. In: 2013 3rd IEEE international advance computing conference (IACC). pp 957–962
6. Zuo L, Shu L, Dong S, et al (2015) Special section on big data services and computational intelligence for industrial systems a multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing. <https://doi.org/10.1109/ACCESS.2015.2508940>
7. Ramezani F, Lu J, Hussain FK, et al (2014) Task-based system load balancing in cloud computing using particle swarm optimization. *Int J Parallel Prog* 42:739–754. <https://doi.org/10.1007/s10766-013-0275-4>
8. Mandal T, Acharyya S (2015) Optimal task scheduling in cloud computing environment: meta heuristic approaches. In: 2015 2nd International conference on electrical information and communication technologies (EICT). pp 24–28
9. Elaziz MA, Xiong S, Jayasena KPN, Li L (2019) Task scheduling in cloud computing based on hybrid moth search algorithm and differential evolution. *Knowl Based Syst* 169:39–52. <https://doi.org/10.1016/J.KNOSYS.2019.01.023>
10. Liu Q, Wei Y, Leng S, Chen Y (2017) Task scheduling in fog enabled Internet of Things for smart cities. In: 2017 IEEE 17th International conference on communication technology (ICCT). pp 975–980
11. Ghobaei-Arani M, Souri A, Safara F, Norouzi M (2020) An efficient task scheduling approach using moth-flame optimization algorithm for cyber-physical system applications in fog computing. *Trans Emerg Telecommun Technol* 31:1–14. <https://doi.org/10.1002/ett.3770>
12. Zitar RA, Al-Betar MA, Awadallah MA et al (2022) An intensive and comprehensive overview of JAYA algorithm, its versions and applications. *Arch Comput Methods Eng* 29:763–792. <https://doi.org/10.1007/S11831-021-09585-8/TABLES/2>
13. Singh P, Chaudhary H (2020) A modified jaya algorithm for mixed-variable optimization problems. *J Intell Syst* 29:1007–1027. <https://doi.org/10.1515/jisys-2018-0273>
14. Chakraborty UK (2020) Semi-steady-state jaya algorithm for optimization. *Appl Sci (Switzerland)* 10. <https://doi.org/10.3390/APP10155388>
15. Awadallah MA, Al-Betar MA, Hammouri AI, Alomari OA (2020) Binary JAYA algorithm with adaptive mutation for feature selection. *Arab J Sci Eng* 45:10875–10890. <https://doi.org/10.1007/s13369-020-04871-2>
16. Satapathy SC, Rajinikanth V (2018) Jaya algorithm guided procedure to segment tumor from brain MRI. *J Optim* 2018:1–12. <https://doi.org/10.1155/2018/3738049>

17. Yu K, Qu B, Yue C et al (2019) A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module. *Appl Energy* 237:241–257. <https://doi.org/10.1016/j.apenergy.2019.01.008>
18. Mohanty S, Patra PK, Ray M, Mohapatra S (2019) An approach for load balancing in cloud computing using JAYA algorithm. *Int J Inf Technol Web Eng* 14:27–41. <https://doi.org/10.4018/IJITWE.2019010102>
19. Wu C, He Y (2020) Solving the set-union knapsack problem by a novel hybrid Jaya algorithm. *Soft comput* 24:1883–1902. <https://doi.org/10.1007/s00500-019-04021-3>
20. Manickam M, Selvaraj S (2019) Range-based localisation of a wireless sensor network using Jaya algorithm. *IET Sci Meas Technol* 13:937–943. <https://doi.org/10.1049/IET-SMT.2018.5225>
21. Gupta S, Agarwal I, Singh RS (2019) Workflow scheduling using Jaya algorithm in cloud. *Concurr Comput* 31:1–13. <https://doi.org/10.1002/cpe.5251>
22. Mishra K, Pati J, Kumar Majhi S (2022) A dynamic load scheduling in IaaS cloud using binary JAYA algorithm. *J King Saud Univ Comp Inform Sci* 34:4914–4930. <https://doi.org/10.1016/j.jksuci.2020.12.001>
23. Jena UK, Das PK, Mishra K, Kabat MR (2021) Improved binary JAYA algorithm-based scheduling dynamic cloud requests for Cloud-based computing. In: 2021 12th International conference on computing communication and networking technologies (ICCCNT). IEEE, pp 1–6
24. Mohanty MN, Das S (2020) Advances in intelligent computing and communication : proceedings of ICAC 2019

# BLOCK-FEMF: Efficient Forensic Evidence Management Framework Using Blockchain Technology



Praveen M. Dhulavvagol, Atrey Mahadev Anagal, Vinayak Sudhakar Kone, Swaroop Anegundi, and Praveen Devadakar

**Abstract** Blockchain technology is a revolutionary, secure, and transparent decentralized digital ledger for recording and storing transactions across multiple computers. It is used to secure, track, and manage digital assets, protecting the data from tampering, revision, and deletion. It provides a secure, immutable, tamper-proof ledger to store digital evidence. Blockchain technology can also track the provenance of digital evidence, ensuring that only authenticated and authorized individuals can access the evidence. Existing evidence management systems lack features like authentic intermediate user access and migrating evidence access from one user to another. The existing system uses the Base32 Algorithm, which lacks storage capability, time delay, scalability, and throughput issues. To overcome these issues, in this paper, we propose a Base64 algorithm integrated with IPFS to enhance transaction performance, scalability, and throughput. The Base64 Algorithm encrypts the image evidence and stores it in the blockchain. The IPFS stores images in a decentralized system to optimize the memory and enhance the throughput of transactions in the blockchain environment. The experimental results show that the proposed Base64 Algorithm optimizes memory utilization by 20% and gas utilization by 19.5% and enhances transaction scalability by 20% compared to the existing Base32 Algorithm.

**Keywords** Blockchain · Ethereum · Base64 · Base32 · IPFS

---

P. M. Dhulavvagol · A. M. Anagal · V. S. Kone (✉) · S. Anegundi · P. Devadakar  
School of Computer Science and Engineering, KLE Technological University, Hubballi,  
Karnataka) 580031, India  
e-mail: [vskonenpn@gmail.com](mailto:vskonenpn@gmail.com)

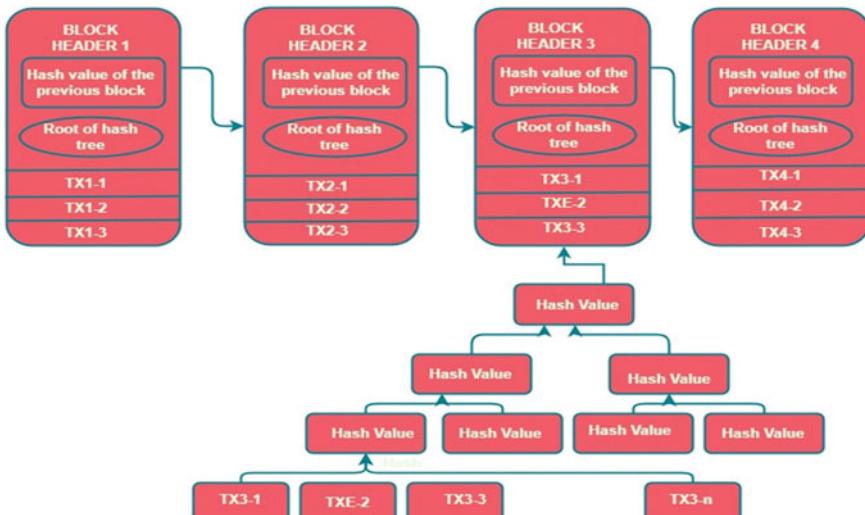
P. M. Dhulavvagol  
e-mail: [praveen.md@kletech.ac.in](mailto:praveen.md@kletech.ac.in)

## 1 Introduction

Blockchain is a decentralized technology that enables the secure transfer of data and assets without the intervention of a trusted third party. It consists of a network of nodes, each of which maintains a copy of a shared ledger of transactions. These transactions are grouped into blocks, which are cryptographically linked and added to the blockchain in a linear, chronological order. One of the key features of blockchain technology is its use of cryptographic algorithms for data security [1]. Each transaction on the blockchain is cryptographically signed by the sender, ensuring that it cannot be altered or deleted without their permission. Additionally, the decentralized nature of the network makes it resistant to tampering, as any attempt to modify the ledger would require the consensus of the nodes in the network. Figure 1 illustrates a configuration of a blockchain network, where interconnected blocks are utilized to create a secure and transparent system.

The applications of blockchain technology are vast and varied, ranging from finance and supply chain management to voting and identity verification. Its unique combination of security, transparency, and decentralization has made it a promising tool for a wide range of industries and use cases. In recent years, there has been a growing demand for blockchain technology for forensic investigations, as it offers a secure and efficient way to collect and transfer evidence in criminal cases [2].

In forensic investigations [3], the collection and transfer of evidence play a crucial role in determining the outcome of a case. Evidence must be collected, preserved, and analyzed carefully to ensure its integrity and reliability. The traditional method of transferring evidence, which involves physically transporting it from the crime scene



**Fig. 1** Blockchain network

to the laboratory, is time-consuming and potentially insecure. A blockchain-based system enables secure and efficient evidence transfer from the crime scene to the laboratory without needing physical transportation [4]. Evidence can be collected, encoded, and added to the blockchain as transactions, ensuring its integrity and authenticity. In addition, the decentralized nature of blockchain technology enables the involvement of multiple parties in the transfer process. For example, law enforcement agencies, forensic laboratories, and judges will be part of the transfer and analysis of evidence without a central authority. This ensures the transparency and accountability of the process and allows for the efficient coordination of resources.

Furthermore, using cryptographic algorithms in a blockchain-based system allows for the secure transfer of sensitive evidence, such as DNA samples and fingerprints. This can help to prevent tampering and ensure the reliability of the evidence, which is crucial for the success of forensic investigations. Further research is needed to explore this technology's potential limitations and challenges and to develop practical applications for its use in forensic investigations.

The main contributions of the proposed work are:

- Analyze and build a smart contract proposal to transfer shreds of evidence between users.
- Design and implement a Base64 algorithm for securing forensic evidence.
- To integrate the Base64 Algorithm with the blockchain PoW consensus protocol to enhance the security and transaction throughput.
- To perform a comparative study analysis of the proposed technique with the current state of art techniques.

The main application of the blockchain forensic evidence management system:

- Blockchain technology can be used to create a tamper-proof chain of custody for digital evidence. This would ensure that the evidence is not altered or tampered with during the investigation and trial process. The immutable nature of the blockchain ensures that any changes to the data are easily detectable.
- Blockchain technology can be used to create digital signatures for evidence. This would ensure that the evidence is authentic and has not been tampered with. The use of digital signatures can help to eliminate the need for physical signatures, which can be forged or altered.
- Blockchain technology can be used to create a timestamp for digital evidence. This would ensure that the evidence was collected at a specific time and date and has not been altered since then. Timestamping can be used to establish the authenticity and integrity of digital evidence.

The paper is organized according to the following sections: Section 1 briefly discusses the introduction of the blockchain network and forensic evidence management methods. Section 2 deals with the existing techniques and literature survey. Section 3 discusses the methodology of the proposed techniques. Section 4 shows the implementation details. Section 5 discusses the results and discussion of the proposed technique in comparison with the existing techniques. Finally, Section 6 shows the conclusion.

## 2 Related Work

The decentralized nature of blockchain technology and its ability to provide secure and tamper-proof data storage make it a promising tool for forensic investigations in the IoT and social systems. Current applications like cybercrime investigation, digital evidence management, and data privacy protection are adopting blockchain technology for secure and faster transaction processing operations. Blockchain technology in forensic investigations can improve the process's security [5], efficiency, and transparency. Additionally, [6] provides a detailed overview of the potential applications of blockchain technology in forensic investigations in the context of the IoT and social systems. In [7], a Hyperledger-based private blockchain network is used to secure medical forensic systems. The existing medical forensic system is subject to various threats, including data tampering, data theft, and data leakage. These threats can lead to severe consequences, such as a breach of patient data confidentiality. Moreover, the current system is inefficient and complex, making it difficult for healthcare professionals to access and share data efficiently. In addition, the blockchain would enable healthcare professionals to access and share data securely and efficiently. Moreover, the paper outlines the process of data storage on the blockchain and the different types of consensus mechanisms that can be used to ensure the integrity of the data stored on the blockchain [7].

The importance of digital image forensic authentication is discussed in [8]. It introduces a practical workflow for digital image authentication and comprehensively evaluates the workflow's effectiveness. The author discusses the need for digital image authentication and details its current challenges. The authors then present the workflow in detail, discussing the various data acquisition and analysis techniques used. Next, the workflow evaluation is discussed, including the results for the four steps of the workflow. The paper then discusses the implications of the workflow for digital image forensics. It explains the importance of having a well-defined workflow to ensure digital images' authenticity.

Combining Artificial Intelligence (AI) and blockchain technologies can create a new forensic investigation tool [9]. This new tool, Forensic Blockchain, aims to improve digital forensics' effectiveness. The authors discuss the current state of digital forensics and the weaknesses that make it challenging to use in many investigations. The authors explain that the Forensic Blockchain is a distributed ledger containing all information related to a particular case. The authors also outline the potential of using AI to detect anomalies in the data that may indicate fraud or other illegal activities. The authors go on to discuss the potential benefits of using Forensic Blockchain. They explain that it can reduce the time and resources needed to complete an investigation while increasing the accuracy of results. Additionally, the authors suggest that it can help reduce the cost of digital forensics, as it eliminates the need for expensive and time-consuming manual processes [8, 9].

An enhancement of the traceability model based on a scenario for the digital forensic investigation process is proposed in [10]. The proposed traceability model is based on a scenario-based approach. This approach is developed to provide the

investigator with an enhanced understanding of the evidence related to a digital forensic investigation. The scenario-based approach is implemented with a case-specific timeline that contains evidence, activities, and tasks associated with the digital forensic investigation process. The proposed traceability model is evaluated using a case study. The evaluation results show that the proposed model effectively and comprehensively provides traceability to digital forensic investigation processes. Furthermore, the evaluation indicates that the proposed model can effectively analyze and document the activities and tasks associated with a digital forensic investigation. The proposed model in [10] provides an enhanced understanding of the evidence related to a digital forensic investigation process and improves the traceability of the activities and tasks associated with the process. The evaluation results demonstrate that the proposed model effectively and comprehensively provides traceability to digital forensic investigation processes.

LEChain [11], a blockchain-based lawful evidence management scheme for digital forensics, is designed to provide a secure and reliable digital evidence storage and management method. Firstly, LEChain provides a secure evidence storage mechanism by utilizing a distributed ledger. This ledger is used to store digital evidence in an immutable and tamper-proof manner. The evidence is stored in a distributed hash table (DHT) secured by a Merkle tree. This ensures that the data stored in the DHT can be verified and traced back to its source. Additionally, LEChain utilizes a consensus mechanism to ensure that all parties involved in the evidence management process agree on the validity of the evidence. Secondly, LEChain utilizes a multi-party computation (MPC) protocol to protect the privacy of digital evidence. This protocol enables the different parties involved in the evidence management process to securely access and use digital evidence without exposing it to third parties. The protocol also ensures that only authorized parties can access the evidence, further enhancing the system's security. Thirdly, LEChain provides a secure evidence access control mechanism. This mechanism allows the different parties to securely access and use the digital evidence without exposing it to third parties or unauthorized users. This mechanism also ensures that access to the evidence is auditable and traceable. LEChain is an excellent digital forensics system that can securely store and manage digital evidence [11].

### 3 Proposed Methodology

The Base64 Algorithm [12] is a widely used method for encoding and decoding data. It represents binary data, such as images and audio files, in an ASCII string format, allowing it to be transmitted over networks that only support text-based data. This is particularly useful for securely and efficiently transfer large amounts of data, such as high-resolution images and video files. The Base64 Algorithm uses a 64-character set, including all upper and lowercase letters, numbers, and special characters. Each group of three bytes of binary data is converted into four characters from the Base64 character set, resulting in a four-fold increase in the encoded data size. The reverse

process, known as decoding, converts the Base64-encoded data back into its original binary form.

One of the key advantages of the Base64 Algorithm is its simplicity and efficiency. It does not require complex mathematical operations, making it suitable for implementation on many devices, including low-power devices such as smartphones and IoT devices. Additionally, the use of Base64 allows for the efficient compression of data, which reduces the amount of time and resources required for its transfer. However, the use of Base64 also has some limitations. The increased size of the encoded data can result in higher storage and bandwidth requirements, which can impact the system's performance.

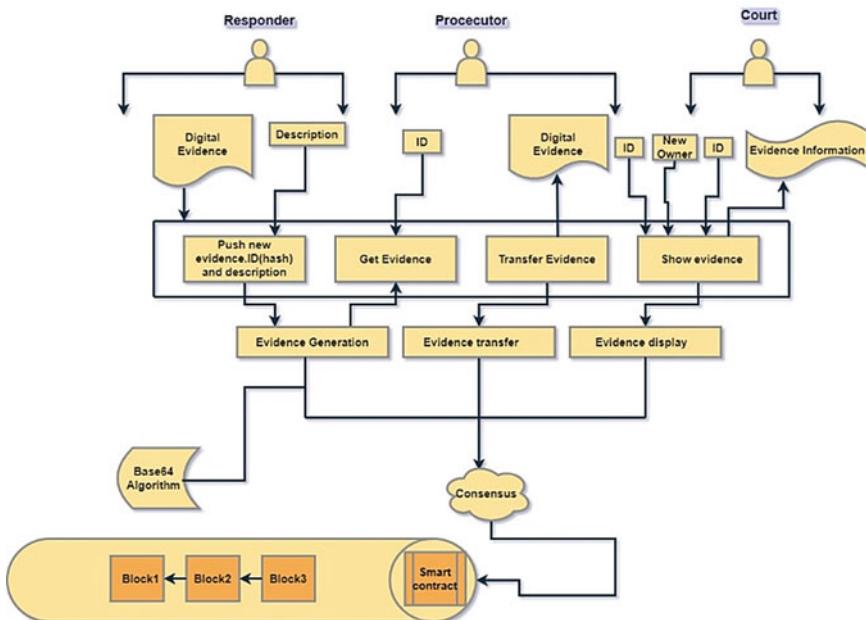
Additionally, using a fixed-length encoding scheme means that some information is lost during the encoding and decoding process, which can affect the quality of the transferred data. The Base64 Algorithm is a widely used and effective data encoding and decoding. Its simplicity and efficiency make it a valuable tool for many applications, including the transfer of forensic evidence in criminal investigations.

As shown in Fig. 2 the proposed system architecture, a decentralized blockchain network serves as the underlying infrastructure for the transfer and storage of forensic evidence. The network would be composed of nodes responsible for verifying and validating transactions and maintaining a copy of the blockchain ledger. A user interface allows forensic investigators, law enforcement agencies, and other relevant parties to access the system and submit evidence. The user interface would provide secure authentication and authorization mechanisms and allow for the creation of transactions that encode the evidence in a standardized format. The evidence encoding and decoding module uses the Base64 Algorithm to encode and decode the forensic evidence. The module would allow for the efficient compression of data, and the secure transfer of sensitive information, such as DNA samples and fingerprints.

A transaction validation and verification module ensure the integrity and authenticity of the transactions on the blockchain network. The module would use cryptographic algorithms and consensus mechanisms to validate and verify transactions and prevent tampering and fraud.

The proposed methodology is divided into mainly four modules as follows:

- (a) **Create user:** Create user functions in the system can be used to create a digital identity for users who wish to access the evidence. This process helps to verify the identity of users and provides them with a unique identifier that is linked to the blockchain. Digital identity helps ensure that the user is who they say they are and can be trusted to access the blockchain. This can help prevent fraud and other malicious activities on the blockchain. This process can also be used to help trace evidence and other data related to the blockchain.
- (b) **Create case:** The case function in the system is used to create a case file for any investigation. This function can store all relevant evidence, documents, and other data related to the investigation. This function can also store the digital evidence and records of the investigation in an immutable, secure, and distributed ledger. This ensures that the evidence will remain intact and unaltered for review and



**Fig. 2** Proposed architecture

analysis. Blockchain technology allows the entire case file to be securely stored, making it impossible for any outside entity to alter or delete it.

- (c) **Get user details:** The get user details function in the system is used to obtain information about a particular user or wallet address on the blockchain network. These data can be used to trace the history of a wallet, identify its owner, and gain insights into its financial activity. By analyzing the data associated with a particular user, digital forensics teams can uncover any suspicious activity that may have occurred. This can help find fraud, money laundering, and other illicit activities. Additionally, digital forensic teams can use this data to provide evidence for related legal proceedings to evidence.
- (d) **Get case information:** The get case information function in the system is a way to retrieve data from the blockchain. These data can be used to help investigators reconstruct events and identify those responsible for criminal activities. For example, the data can be used to track the flow of assets between addresses, identify the time and place of transactions, and determine the identity of the participants. Additionally, these data can be used to uncover patterns of behavior and establish links between entities.

## 4 Implementation

### 4.1 Adding Users to the Blockchain Network

The append into chain function in the system is a process that allows new data to be added to an existing blockchain network. This process is used to help maintain the integrity of the data stored on the blockchain by ensuring that all transactions are recorded and stored in a secure and immutable manner. The process of appending data to the blockchain is done by verifying the data, encrypting it, and then adding it to the chain. This process helps to ensure that the data stored on the blockchain are accurate and secure.

---

**Algorithm 1: Adding Users to the blockchain network**

---

```
FUNCTION AppendIntoChain(owner_Id,user_Id,case_Id){
    IF owner_Id exists and has ownership of case_Id
        IF user_Id exists
            insert case_Id into On_Chain_Cases of user_Id
            print "User is appended into Chain"
            return true
        ELSE
            print "Invalid User_ID"
            return false
    ELSE
        print "Invalid Ownership_ID"
        return false
}
```

### 4.2 Transfer Ownership

Transfer ownership in the system is a process that provides an additional layer of security for digital assets. It allows for the transfer of ownership of digital assets from one party to another in a secure and immutable fashion. This process is done through digital signature technology, which provides a cryptographically secure way to transfer ownership of assets from one party to another. The blockchain allows for an immutable record of the transfer of ownership, providing evidence for digital forensics investigators in the event of a dispute or investigation. This technology also allows for faster, more efficient transfers of ownership, reducing the need for manual paperwork and other processes.

---

**Algorithm 2: Ownership Transfer**

---

```

FUNCTION Transfer_Ownership(owner_Id,user_Id,case_Id){
    IF owner_Id exists and has ownership of case_Id
        IF user_Id exists
            remove ownership of owner_Id for case_Id
            add ownership of user_Id for case_Id
            print "Ownership Transferred from owner_Id to user_Id"
        ELSE
            print "Invalid User_ID."
    ELSE
        print "Invalid ownership_ID."
}

```

### 4.3 Adding Evidence to the Case

Insert evidence is a function in the system that allows investigators to store digital evidence securely on a blockchain. This evidence can then be used to prove the veracity of a digital investigation or audit. The evidence stored on a blockchain is immutable and tamper-proof, meaning that it cannot be changed or manipulated by anyone, making it an ideal solution for forensics evidence management. Blockchain technology also offers a secure and transparent way to store digital evidence, providing investigators with a secure source of information that can be used in court. Insert evidence can be used to store any type of digital evidence, including photos, documents, audio, video, and more. The use of blockchain technology in digital forensics has the potential to revolutionize the way digital evidence is gathered and used in investigations, providing a secure and transparent way to ensure the accuracy of digital investigations.

---

**Algorithm 3: Adding evidence to the case**

---

```

FUNCTION Insert_Evidence(case_Id,evidence_Id,user_Id,evidence){
    IF case_Id exists
        IF userId exists and userId has ownership of case_Id
            IF evidence_Id does not exists
                insert evidence into evidence_Id under case_Id
                print "Evidence Inserted Successfully"
            ELSE
                print "Invalid Evidence ID"
            ELSE
                print "Invalid User ID"
        ELSE
            print "Invalid Case ID"
}

```

#### **4.4 Retrieving of Evidence from Case File**

Get evidence information function in the system allows forensic investigators to obtain evidence from the blockchain to investigate any suspicious or unlawful activities. It helps in the collection, analysis, and authentication of digital evidence. It includes retrieving transaction records, public and private keys, addresses, and other blockchain data. It also allows investigators to trace the flow of digital assets, identify the parties involved in a transaction, and establish the time and origin of any digital asset. Blockchain technology can also create an immutable digital evidence ledger that can be used in legal proceedings. These functions allow investigators to identify criminal actors, trace transactions, and aid in investigating crimes.

---

**Algorithm 4: Retrieving of evidence from case file**

---

```

FUNCTION Get_Evidence_Information(case_Id,evidence_Id,user_Id){
    IF case_Id exists
        IF userId exists and userId is in chain of case_Id
            IF evidence_Id exists
                print "Evidence Information Fetched Successfully"
                return evidence details of evidence_Id under case_Id
            ELSE
                print "Invalid Evidence ID"
            ELSE
                print "Invalid User ID"
        ELSE
            print "Invalid Case ID"
    }
}

```

## **5 Results and Discussions**

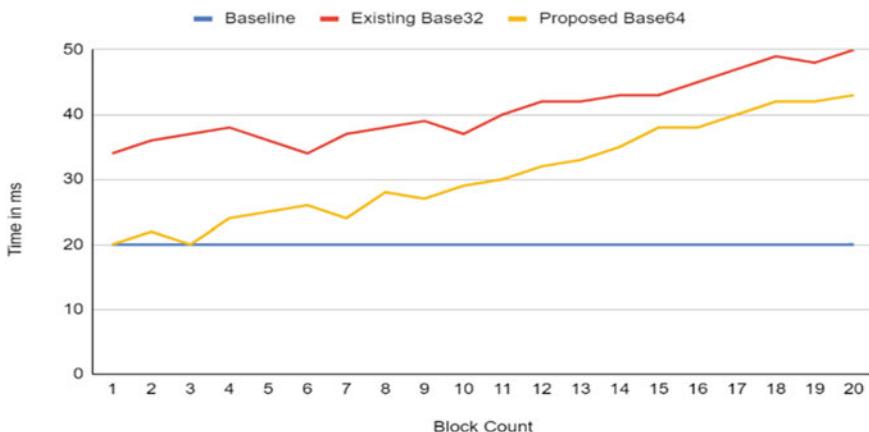
The experiment is conducted considering Core i5-1035G1 processor, NVIDIA GeForce MX250 graphics with 2 GB DDR5 and 8 GB of DDR4-2666 SDRAM memory, with Windows 11. The dataset is considered from stegoAppDB [13]. The dataset is made up of a selection of photographs that were gathered from numerous websites, social media sites, and mobile apps, among other places. The least significant bit (LSB), discrete cosine transforms (DCT), and spread spectrum (SS) are among the steganography techniques used to embed the secret message in the photos in the dataset. As a control group, the collection also contains photographs that were not altered using steganography. 6,800 total photos make up the dataset, 5,000 of which were intentionally altered using steganography methods. Sample images of 500 MB in size with 30 blocks were considered for evaluation of the proposed technique.

## 5.1 Gas Value Consumption

In the context of the blockchain, gas refers to the unit measuring the amount of computational effort required to execute a particular operation or transaction on the Ethereum network. The gas consumption is directly related to the number of blocks added to the blockchain, as each block requires a certain amount of computational effort to be added to the chain. As the number of blocks increases, the amount of gas consumed increases, as more computational effort is required to add additional blocks to the blockchain. Therefore, the consumption of gas value is shown to increase gradually as the number of blocks in the blockchain increases in Fig. 5.4.

## 5.2 Time Consumption

The time required to create a new block on the blockchain can vary depending on several factors, including the current level of network activity and the complexity of the transactions being processed in each block. With the increasing number of users and transactions, the blockchain network expands and requires more time to create a new block, which requires more gas value. The network latency tends to decrease, affecting the transaction scalability, as shown in Fig. 3. The existing Base32 algorithm takes more gas value than the proposed Base64 Algorithm for block creation and processing operation. However, the exact relationship between the size of the blockchain and the time it takes to create a new block can vary and is not necessarily a linear relationship.



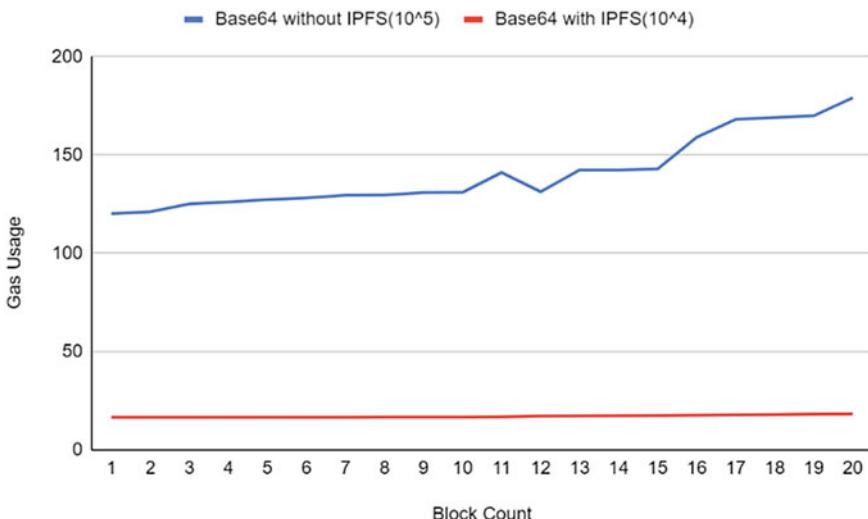
**Fig. 3** Graph showing proposed Base64 work versus existing Base32 work's time consumption with baseline reference for the creation of every new block

In Fig. 3, block count is represented along the x-axis and time along the y-axis. The current work used the Base32 Algorithm, which uses 20% more size than the Base64 Algorithm; since the time is based on the data size, the time increases as the size increases. Hence, the proposed model uses more time when compared to theoretical results. The curve in blue represents the proposed work, and the curve in red represents the existing work.

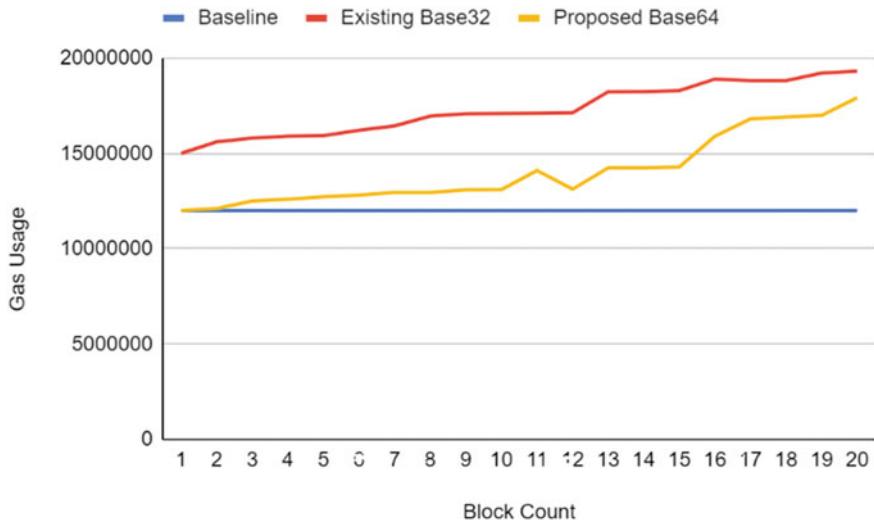
In Fig. 4, block count is represented along the x-axis, and gas price utilization along the y-axis. The current work used the Base32 Algorithm, which uses 20% more size than the Base64 Algorithm; since the gas price is based on the size of the data, the gas price utilization increases as the size increases. Hence, the proposed model uses less gas when compared to an existing model. The curve in blue represents the proposed work, and the curve in red represents the current work.

In Fig. 5, the blue curve represents Base64 without IPFS, whereas the red line indicates Base64 with IPFS. Here, it is possible to see that on the x-axis, we have a block count, and on the y-axis, gas usage. From the nature of the graph, it is possible to predict that as the block creation increases drastically without the use of IPFS, the gas taken to insert the same image increases. This means that there is an issue of scalability; as we add more images into the blockchain environment, the gas usage increases, hence increasing the use of more Ethereum. Therefore, we are integrating Base64 with IPFS to solve the scalability issue, the graph that can be seen in the red line.

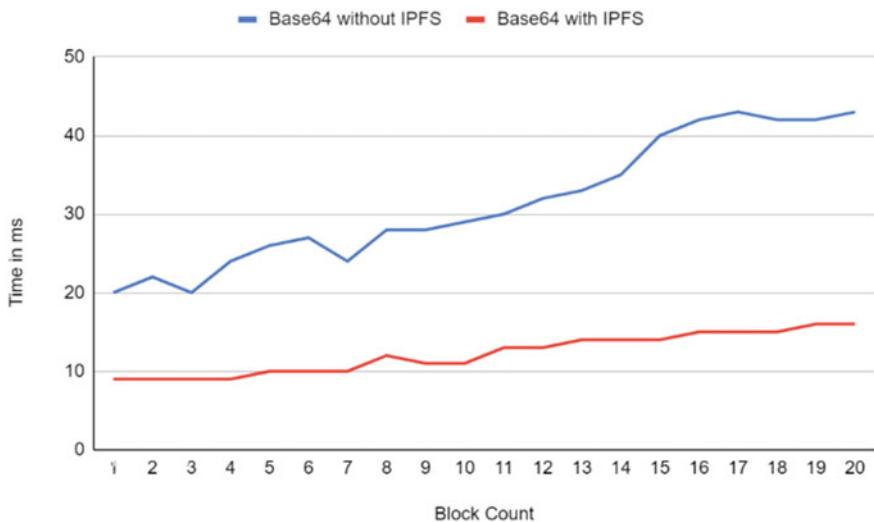
In Fig. 6, the blue curve represents Base64 without IPFS, whereas the red line indicates Base64 with IPFS. Here, we can see that on the x-axis, we have block count, and on the y-axis, time taken. From the nature of the graph, it is possible to



**Fig. 4** Graph showing proposed Base64 work versus existing Base32 work's gas price utilization with baseline reference for the creation of every new block



**Fig. 5** Graph showing Base64 with IPFS vs Base64 without IPFS gas price utilization for the creation of every new block



**Fig. 6** Graph showing Base64 with IPFS vs Base64 without IPFS time consumption for the creation of every new block

predict that as the block creation increases, the time taken to insert the same image also increases without integrating with IPFS. This means that there is an issue of time complexity; as we go on adding more images into the blockchain environment; the time taken goes on increasing hence increasing the time taken to insert images. Therefore, we are integrating Base64 with IPFS to solve the scalability issue, the graph that can be seen in the red line.

One possible mathematical model for blockchain-based evidence management could involve a probabilistic graph model that represents the various pieces of evidence and their relationships. The graph could consist of nodes representing individual pieces of evidence, such as timestamps, digital signatures, and document hashes, along with edges representing the relationships between them. For example, an edge could connect two timestamps if they are part of the same chain of custody, or between a timestamp and a digital signature if the signature verifies the authenticity of the timestamp.

Using this model, we can generate hypotheses about the authenticity and verifiability of evidence. For example, we could hypothesize that if a timestamp is connected to a digital signature by a chain of custody edges, then the evidence is highly likely to be authentic and verifiable. Alternatively, if a document hash is connected to multiple conflicting timestamps, then the evidence may be unreliable and require further investigation.

### 5.3 Comparative Study Analysis

Table 1 compares the existing Base32 Algorithm to the proposed Base64 and Base64 + IPFS algorithms. The table lists various parameters for each algorithm, such as the average gas consumption for 20 kb images, the average time for block creation in milliseconds, the gas usage reduction percentage when compared to the Base32 algorithm, and the throughput/scalability in transactions per second. For 20 kb images, the Base64 algorithm has an average gas consumption of 1,20,71,102.85 and a block creation time of 31.5ms. It also reduces gas usage by 19.56% when compared to the Base32 algorithm and has a throughput/scalability of 2T/s. In comparison, the Base64 + IPFS algorithm has an average gas consumption of 1,67,144.67 for 20 kb images, with a block creation time of 12.4ms. It reduces gas usage by 98.98% when compared to the Base32 algorithm and has a higher throughput/scalability of 5T/s.

Table 2 summarizes various privacy and security features for various technologies such as PoC in Hyperledger [14], Process Provenance [15], Anonymous Witnessing [16], Block-DEF [17], Identity privacy [18], and the proposed Base64 + IPFS algorithm. The table shows the various properties of each technology, such as authentication, access control, intermediate nodes, transferring ownership, and integrity.

In several models, various blockchain technologies have been utilized to implement different features for securing and tracking data. PoC in Hyperledger [14] uses

**Table 1** Comparative study table of existing Base32 algorithm with proposed Base64 and Base64 + IPFS algorithm

Algorithm	Parameters			
	Average gas consumption	Average time for block creation	Gas usage reduction (%)	Scalability
Base 64	1,20,71,102.85	31.5ms	19.56	2T/s
Base 64 + IPFS	1,67,144.67	12.4ms	98.98	5T/s

**Table 2** Comparative study analysis of proposed Base64 + IPFS algorithm versus existing algorithms

Property	PoC in Hyperledger [14]	Process provenance [15]	Anonymous witnessing [16]	Block-DEF [17]	Identity privacy [18]	Proposed Base64 + IPFS
Authentication	Yes	Yes	Yes	Yes	Yes	Yes
Access control	Yes	No	No	No	Yes	Yes
Intermediate nodes	Yes	No	No	No	No	Yes
Transferring ownership	No	No	No	No	No	Yes
Integrity	Yes	Yes	Yes	Yes	Yes	Yes

a chain of custody using Hyperledger, which included access control and intermediate nodes. However, the transferring ownership feature could not be implemented. Similarly, Process Provenance [15] is used to implement cloud forensics, which included only authentication, but the transferring ownership, access control, and intermediate nodes features could not be implemented. In Anonymous witnessing [16] a digital chain of custody in IoT is implemented, including authentication and integrity features. However, the transferring ownership, access control, and intermediate nodes features were not implemented. Block-DEF [17], included authentication and integrity. However, transferring ownership, access control, and intermediate nodes features were not implemented. Lastly, Identity privacy is used to implement a chain IoT forensic framework, which included access control, authentication, and integrity features. However, the transferring ownership and intermediate node features were not implemented.

## 6 Conclusion

Blockchain is being used in forensic evidence management to have a secure transfer of digital evidence between the users. The existing system uses Base32 algorithm, which lacks in transaction scalability and consumes more gas value, to overcome these limitations in this paper we propose a Base64 Algorithm integrated with IPFS.

To optimize the memory utilization of blockchain network IPFS is integrated with the blockchain network. The experiment results show that the proposed Base64 Algorithm optimizes the memory storage by 20% compared to the existing Base32 Algorithm, and the transaction latency is reduced by 5% compared to Base32 algorithm. By storing only, the hash value of data in the blockchain network and IPFS and keeping the real data kept elsewhere, our solution reduces storage by 20%. Additionally, using IPFS allows the blockchain network to reduce storage by 90%. We further examine the Base32 and Base64 encoding methods' time effectiveness and demonstrate that Base64 encoding can shorten the time by 31.1%. Base64 encoding and IPFS work together to significantly cut down on processing time (72.74%). Applications where storage, transaction latency, and time effectiveness are important considerations may benefit from our approach. The gas value is reduced by 19.56%. The proposed approach Base64 integrated with IPFS enhances the transaction scalability and reduces the gas utilization by 98.98%. Thus, integrating with IPFS made the proposed model more efficient and cost-effective and resolved the issue of memory, higher time delay, lower scalability, and more gas utilization.

## References

1. Dhulavvagol PM, Bhajantri VH, Totad SG (2020) Blockchain ethereum clients performance analysis considering E-voting application. Proc Comp Sci 2020 Author F, Author S (2016) Title of a proceedings paper. In: Editor F, Editor S (eds) Conference 2016, LNCS, 9999:1–13. Springer, Heidelberg
2. Chang SE, Chang C-Y (2018) Application of blockchain technology to smart city service: a case of ridesharing. In: 2018 IEEE International conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData). IEEE
3. Mansourvar M, et al (2012) A computer-based system to support intelligent forensic study. In: 2012 Fourth international conference on computational intelligence, modelling and simulation. IEEE
4. Wang W, et al (2019) A survey on consensus mechanisms and mining strategy management in blockchain networks. IEEE Access 7:22328–22370
5. Wang D, Zhang X (2020) Secure ride-sharing services based on a consortium blockchain. IEEE Internet Things J 8(4):2976–2991
6. Li S, Qin T, Min G (2019) Blockchain-based digital forensics investigation framework in the internet of things and social systems. IEEE Trans Comput Soc Syst 6(6):1433–1441
7. Ahmed M, et al (2020) Securing forensic medical system using hyperledger based private blockchain. In: 2020 23rd International conference on computer and information technology (ICCIT). IEEE
8. Zeng J, et al (2016) Practical inspection workflow for digital image forensic authentication. In: 2016 4th International symposium on digital forensic and security (ISDFS). IEEE
9. Mani N, et al (2021) Forensic block chain and it's linkage with artificial intelligence: a new approach. In: 2021 2nd International conference on computation, automation and knowledge management (ICCAKM). IEEE
10. Mohamed IA, Manaf ABA (2014) An enhancement of traceability model based-on scenario for digital forensic investigation process. In: 2014 Third International conference on cyber security, cyber warfare and digital forensic (CyberSec). IEEE

11. Li M, et al (2021) LEChain: a blockchain-based lawful evidence management scheme for digital forensics. *Fut Generat Comp Syst* 115:406–420
12. Guwalani P, et al (2014) Image file security using base-64 algorithm. Pooja Guwalani *Int Comput Technol Appl* 5:1892–1895
13. Newman J, et al (2019) Stegoappdb: a steganography apps forensics image database. arXiv preprint [arXiv:1904.09360](https://arxiv.org/abs/1904.09360)
14. Lone AH, Mir RN (2019) Forensic-chain: blockchain based digital forensics chain of custody with PoC in hyperledger composer. *Digit Invest* 28(2019):44–55
15. Zhang Y, Wu S, Jin B, Du J (2017) A blockchain-based process provenance for cloud forensics. In: Proceedings of the 3rd IEEE International conference on computer and communications. ICCC, Chengdu, China, pp 2470–2473
16. Nieto A, Rios R, Lopez J (2017) Digital witness and privacy in IoT: Anonymous witnessing approach. In: Proceedings of the 16th IEEE International conference on trust, security and privacy in computing and communications. TrustCom, Sydney, Australia, pp 642–649
17. Tian Z, Li M, Qiu M, Sun Y, Su S (2019) Block-DEF: a secure digital evidence framework using blockchain. *Inform Sci* 491:151–165
18. Le D-P, Meng H, Su L, Yeo SL, Thing V (2018) BIFF: a blockchain-based IoT forensics framework with identity privacy. In: Proceedings of the international technical conference of IEEE region 10. TENCON, Jeju Island, Korea, pp 2372–2377

# Bacterial Spot Classification in Bell Pepper Plants Based on Convolutional Neural Network Architecture



Midhun P. Mathew, Sudheep Elayidom, and V. P. Jagathiraj

**Abstract** In the field of agriculture, classification of disease is an important problem that needs more attention. There are many undergoing projects for the classification of diseases in plants. The traditional methods of classification require more effort and time at the same time the accuracy is too low, this leads to the problem in the production of yield. Nowadays, classification of diseases using computer vision gains so much attention among the government, researchers and farmers. The classification of diseases using computer vision technology provides an automation solution for the loss of yield. Identification of disease at its initial stage is very important to reduce the loss of yield. On the bell pepper leaves, machine learning and deep learning approach help to classify the diseases at its initial stage. There are so much research undergoing on machine learning for disease identification in plants, such as tomato, potato, apple and grapes. But the accuracy of this architecture is less as compared to deep learning architecture. This paper tries to explain the architecture of CNN such as Google Net, ResNet 18, ResNet 50 and Resnet101. This paper also focuses on the performance of Google Net, ResNet 18, ResNet 50, and ResNet 101 based on the classification of disease in bell pepper plants. Based on our study in classification of disease in bell pepper, we found out that Google Net shows better performance as compared to other architecture.

**Keywords** Bell pepper · CNN architecture · Disease classification · Google Net · ResNet 18 · ResNet 50 · ResNet101

---

M. P. Mathew (✉) · S. Elayidom  
CS Division SOE CUSAT, Kochi, India  
e-mail: [midhunpmathew@cusat.ac.in](mailto:midhunpmathew@cusat.ac.in)

V. P. Jagathiraj  
SMS CUSAT, Kochi, India

## 1 Introduction

Indian economy is mainly dependent on the agricultural sector, to develop the country's economy, agricultural production should be increased [1]. So, the government is mainly focused on increasing the production of yields [1]. But the drawback the agriculture sector now facing is disease identification at its right time. Due to this, farmers are facing a huge loss in their production. The plant leaves are mainly affected by climatic changes, fungi, and bacteria. Early identification of the disease will help the farmers to avoid large loss in their yield. In the majority cases, the farmers won't able to find the disease at its early stage. In some cases, farmers will identify the pattern changes in the leaves, then they will either take an opinion of an expert or they will simply assume it as some disease and start doing prevention techniques, but this will not help the farmers to overcome the disease or sometimes it may cause harm to the plant itself. In this scenario, the automation in disease classification plays a major role. The introduction of automation technology can help the farmers to identify the disease at its early stage and also help them to identify the correct disease. The paper introduces a study of different architectures of convolutional neural networks and also identifies the better performance for the classification of the disease in bell pepper. This paper focuses on identification of bacterial spots on the healthy leaves of bell pepper plants using three different CNN architectures such as Google Net, ResNet 18, 50 and 101. This paper also helps to analyze the performance of the CNN architecture such as Google Net, ResNet and its three different versions. The Input Image that we give will pass through the three CNN architectures and classify whether the given image is affected by bacterial spot or not. Here, we use the dataset from plant village, which is available in Kaggle, as the training dataset and tested with images of bell pepper leaves which is available in the Google, Google Image as the tested dataset. The methodology of the work can be classified as follows, first pre-processing of the dataset, followed by the training of the CNN model such as ResNet and Google Net and finally the classification of the model which is done using Google Net and ResNet. The Google Net and ResNet consist of convolutional layer, pooling layer, activation layer, SoftMax layer and fully connected layer. The detailed analysis of each CNN architecture will be explained in the following section.

The paper is organized as the following sections: Section 1 explains about the introduction, Sect. 2 explains about the study of different work which was done in early time, Sect. 3 explains about the transfer learning, Sect. 4 explains the architecture study of Google Net and ResNet, and Sect. 5 explains about the methodology of the work which we have done and, in this section, we explain about the performance of the CNN architecture based on precision, recall, specificity and the accuracy, and also the confusion matrix is plotted in this section, and the result analysis is done. The last section gives the conclusion and future scope of the model.

## 2 Literature Review

Disease identification or classification is the major area of research, in the field of image processing and computer vision. In this section, we will discuss the most common techniques that are used in classification of disease. Paper [2] mentions classification of tomato leaves as healthy or not. Dataset is pre-processed by removing the background and noise. The noise content and background of the image are removed using erosion technique. The feature extraction of the input image is done by using GLCM (Gray Level Co-occurrence Matrix). The proposed work was trained by using a Support vector machine (SVM) classifier and performance was evaluated by using N-fold cross-validation technique. According to the paper, the accuracy of the model is high, but it could not classify which type of disease. To eliminate the problem of [2], paper [3] mentions various methods of feature extraction and classification. Here, the image given as input is pre-processed by histogram equalizer. To identify the affected part of the leaves, different segmentation techniques are used. After that, the extracted features are calculated by using GLCM and using artificial neural network, the disease part is identified. The main drawback of the paper is that it uses clustering, as a result the proposed work is semi-automated. Paper [4] mentions about the classification of disease in tomato, here the segmentation of image is done by using Otsu's threshold approach. The color, texture and shape of the dataset are extracted using different methods, which are combined to create a module of all features. Classification of the disease is done using a decision tree classifier. The paper says that the decision tree classifier provides high accuracy in the classification of disease, but it faces the drawback of overfitting. Paper [5] mentions the convolutional neural network for the classification of diseases. The paper mentions the concept of deep learning and generates a model that is able to identify the disease with high accuracy as compared to machine learning algorithms. Paper [6] mentions disease detection using convolutional neural network architecture LeNet, the proposed architecture achieves high classification accuracy. Paper [7] focuses on the classification of decay in apple trees. In this paper, authors mention the shallow method and CNN method to classify the decay. Paper uses Alex Net for CNN method and multilayer perceptron for shallow method and finds out that CNN with Alex Net has high accuracy as compared to other methods. Paper [8] mentions the importance of transfer learning in disease classification. Paper uses a pre-trained model of Alex Net and VGG 16 Network for the classification of disease, here the image is augmented and the paper concludes that the performance of architecture is better when we increase the image number and also it concludes that the pretrained model of Alex Net shows better performance as compared to VGG 16.

### 3 Transfer Learning

Transfer learning is defined as a creation of a model for one task that can be used for similar tasks. Transfer learning concept will help to make the training process more efficient, easy and faster as compared to the development of a new model from the Scratch. The transfer learning concept is mainly used for applications such as object recognition, object detection and speech recognition. Nowadays, transfer learning becomes more popular because of the following reasons:

- It helps to train a model with less labeled data.
- Training time and computational cost can be reduced.
- Make use of popular architecture of deep learning.

There are various pretrained models available, each model has its own advantages and disadvantages. Following factors should be considered before choosing pretrained models:

- (i) Size: When we are doing a project, we should have a good understanding about the deployment of the project. When we choose an architecture, we should make sure that the concerned architecture will run on embedded hardware or not. In the case of low-memory application, size of the pretrained architecture is very important,
- (ii) Accuracy: Architecture that is performed well in a large dataset will perform well in a new dataset. If the performance of the architecture is low in the case of a large dataset, it may or may not work well on a new dataset. So choose the pretrained architecture based on the requirement.
- (iii) Prediction speed: How fast the network can predict on a new dataset. Prediction speed can vary based on the hardware, batch size, and architecture including its size.

## 4 Google Net and ResNet Architecture

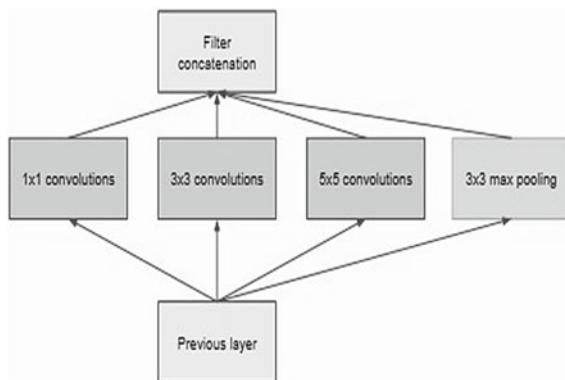
### 4.1 *Google Net*

Paper [9–12], gives a detailed explanation of the Google Net architecture. In deep learning architecture, emergence of Inception network gives a breakthrough to convolutional neural network. Inception network consists of three different versions such as version 1, 2 and 3. Among these versions, the first version is known as Google Net. Google Net is developed by the researchers of Google. Implementation of Google Net helps to implement the new state of classification in convolutional architecture. Suppose if we are building an architecture with a different layer of deep net, then there occurs a fitting issue. To solve the problem of fitting, Google Net architecture uses different sizes of filters, which can work at the same level of the architecture. Thus, Google Net Architecture became wider. Figure 1 shows an inception module

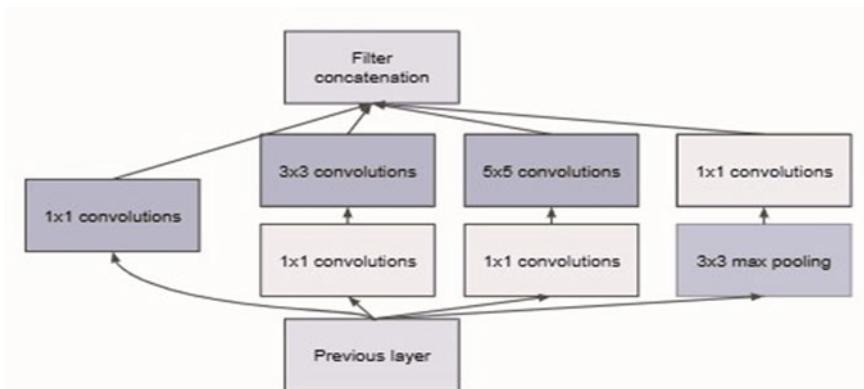
with a naive version. From Fig. 1, we can see that the input image undergoes convolutional operation of three different filter sizes such as one, three and five. Along with this convolutional operation, it moves to max pooling, after that it goes to the next inception layer.

If we train the Navie-inception module, we can see that the training process requires a large amount of time, as a result it becomes more expensive. So, to avoid this problem Google Net architecture developer uses additional  $1 \times 1$  convolutional layer before  $3 \times 3$  and  $5 \times 5$  convolutional layer. It helps to reduce dimensionality of architecture and also training process becomes faster. Figure 2 shows the addition of convolutional layer in Navie-Inception Module.

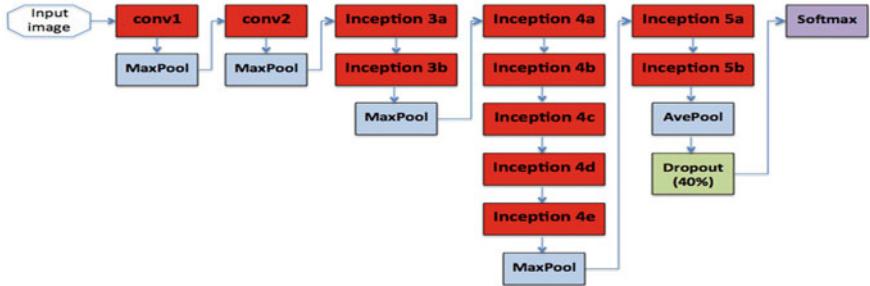
In general, Google Net consists of 22 depth layers along with 27 pooling layers and nine inception modules, these layers are connected to global average pooling, and it also includes four max pooling layers. Figure 3 shows the architecture of Google



**Fig. 1** Naïve-inception module taken from [10]



**Fig. 2** Navie-Inception module with  $1 \times 1$  convolutional layer, building block of Google Net taken from [10]



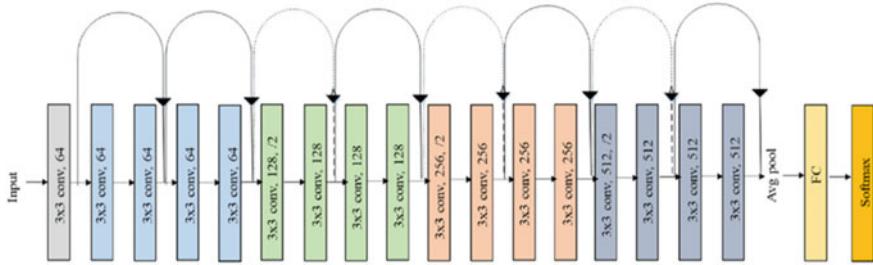
**Fig. 3** Google Net architecture taken from [13]

Net. The Activation function used here is ReLu and is included in the inception layers also. A dropout percentage of 40 is added to the SoftMax layer. In Google Net, the first convolutional layer consists of a filter size of 7. It is used for resizing the input image without losing the spatial content of the image. This is done with the help of 7\*7 filter size. In the second convolutional layer, height and width of an image are adjusted. Second convolution layer has a depth of two and a convolutional layer of 1\*1, helps in dimensionality reduction. There are nine inception modules present in the Google Net, and two max pooling layers presented between some inception modules. These max pooling layers downsample the input image. Downsampling the input image is done by reducing the high and width of the input image.

The Average Pooling layer calculates the mean average from the feature map, which is obtained from the last inception layer and reduces the size of the feature to one. The dropout used in the Google Net is 40%, which helps to prevent the model from overfitting. After the dropout layer, the next layer is the linear layer, which consists of 1000 classes. SoftMax layer takes the functions of the activation layer and is used to generate a set of numbers within an input vector. In the Google Net structure, there is an additional layer added by the developer which is known as the Auxiliary classifier. Auxiliary classifiers consist of the average pool layer, convolutional layer, two fully connected layers, dropout layer and SoftMax layer. These classifiers are added to the intermediate layer of the Inception. The function of the Auxiliary classifier is to perform classification in the middle section of the architecture, it also calculates the loss during training. After training, Auxiliary classifier is removed.

## 4.2 ResNet

Paper [9, 14] mentions ResNet Architecture. The aim of the ResNet is to introduce a shortcut connection that helps to solve the problem based on vanishing gradient. This is done by reusing the activation of the previous layer by skipping the connection. This skipping connection makes the architecture compress, as a result, the learning



**Fig. 4** ResNet 18 Layer structure taken from [15]

capability of the architecture increases. After the compression, there occurs an expansion of the layer, as a result the residual part of the architecture extracts more features. There are different types of ResNet versions such as 16, 18, 34, 50, 101, 110, 152, 164 and 1202. Here, we will explain more about the ResNet 18, 50 and 101.

### ResNet 18

Paper [15] explains about the ResNet18 architecture. Figure 4 shows the layers of ResNet18, from the figure we can see that the ResNet18 consists of 17 convolutional layers, one fully connected layer, and a SoftMax layer. These layers present in the architecture help to classify the image.

Resnet 18 uses  $3 \times 3$ . With three filters in the convolutional layer, the architecture is designed such that the output feather map is the same when the filter number is the same, but when the filter number is double then the feature map of the output layer is reduced to half. In ResNet 18, downsampling is performed by the convolutional layer with stride equal to two. After the convolutional layer, the next layer is the average pooling layer, followed by a fully connected layer and SoftMax layer. From the figure, we can see that there are two connections between the layers, one is solid and the other is a dotted line. The solid line is used when the input and output features have the same dimension, and the dotted line is used when the dimension is increased.

### ResNet 50

ResNet 50 consists of 48 convolutional layers, one average pooling layer and one max pooling layer. In ResNet 50,  $1 \times 1$  convolutional layer is added for the development of the building block. This additional layer of convolution is known as the bottleneck layer, using this layer will help to reduce the parameter and make the architecture faster, it consists of a cluster of three layers of convolution. The following Fig. 5 shows the features of ResNet 50 architecture along with other versions. From Fig. 5, we can see that the ResNet 50 consists of convolutional layer of size  $7 \times 7$  with 64 kernels and a stride of two, after the convolutional layer, then there is a maxpool layer of stride two. Followed by this, there are another nine more layers with  $3 \times 3$ , 64 kernels,  $1 \times 1$ , 64 kernels and  $1 \times 1$ , 256 kernels, which are repeated three times. After this, there are another 12 convolutional layers with  $1 \times 1$ , 128 kernels,  $3 \times 3$ , 128 Kernels and  $1 \times 1$ , 512 kernels repeated four times. Then another 18 convolutional

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112			7×7, 64, stride 2		
				3×3 max pool, stride 2		
conv2.x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3.x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4.x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5.x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1			average pool, 1000-d fc, softmax		
FLOPs	$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$	

**Fig. 5** Element of the ResNet Architecture of different versions taken from [16]

layers consist of 1\*1,256 kernels, 3\*3, 256 kernels and 1\*1, 1024 kernels that are repeated six times. Lastly, another nine convolutional layers are repeated three times, these nine convolutional layers consist of 1\*1, 512 kernels, 3\*3, 512 kernels and 1\*1, 2048 kernels. Thus, a total of 49 convolutional layers are presented in the ResNet 50. After these convolutional layers, there is an average pooling layer, fully connected layer with 1000 nodes (since ResNet 50 is a pretrained model with 1000 class) and finally a SoftMax. Thus, make the 50-layer architecture of ResNet, called ResNet 50.

### ResNet 101

ResNet 101 consists of 99 convolutional layers, one average pooling layer and one max pooling layer. In ResNet 101, 1\*1 convolutional layer is added for the development of the building block. This additional layer of convolution is known as the bottleneck layer, using this layer will help to reduce the parameter and make the architecture faster. It consists of a cluster of three layers of convolution, Fig. 5 shows the features of ResNet 101 architecture along with other versions. From Fig. 5, we can see that the ResNet 101 consists of convolutional layer of size 7\*7 with 64 kernels and a stride of two, after the convolutional layer, then there is a maxpool layer of stride two. Followed by this, there are another nine more layers with 3\*3, 64 kernels, 1\*1, 64 kernels and 1\*1, 256 kernels that are repeated three times. After this, there are another 12 convolutional layers with 1\*1, 128 kernels, 3\*3, 128 kernels and 1\*1, 512 kernels repeated four times. Then another 18 convolutional layers consist of 1\*1, 256 kernels, 3\*3, 256 kernels and 1\*1, 1024 kernels, which are repeated 23 times. Lastly, another nine convolutional layers are repeated three times, these nine convolutional layers consist of 1\*1, 512 kernels, 3\*3, 512 kernels and 1\*1, 2048 kernels. Thus, a total of 99 convolutional layers are presented in the ResNet 101. After these convolutional layers, there is an average pooling layer, fully connected layer with 1000 nodes (since ResNet 101 is a pretrained model with 1000 class) and

finally a SoftMax. These layers make the 101-layer architecture of ResNet, called ResNet 101.

## 5 Methodology of the Work

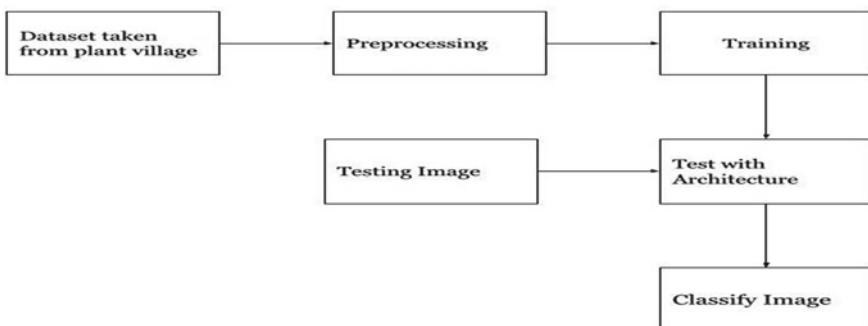
The methodology of the work is explained as follows: Aim of the project, procedure of the work and result generated. Figure 6 shows the general block diagram of the project.

### 5.1 Aim of the Project

The project focuses on the classification of disease in bell pepper leaves and helps the bell pepper farmers to easily identify the disease in their plants with high accuracy. This project work helps to understand the importance of transfer learning and also aims to study the different pre-trained convolutional architecture such as Google Net, ResNet 18, 50 and 101 and identify which model shows better performance in classification of diseases in the bell pepper plants.

### 5.2 Procedure of the Work

As we already mentioned that one of the major aims of the research work is to identify which architecture is more suitable for the disease classification in bell pepper, in this section, we will explain the procedure for the classification of the disease in bell pepper.



**Fig. 6** Block diagram of the proposed work



**Fig. 7** Sample dataset

### Dataset collection

Collection of primary datasets is a difficult task as we have to go to the field and collect data from the field, which consumes a large amount of time and effort. For the proposed work, we choose the dataset that is available in the Kaggle site, plant village. Figure 7 shows the sample dataset of image, which includes Bacterial spot and healthy leaves of bell pepper. We use MATLAB 2022b for doing our work and our work is done on single-core CPU machine.

### Data pre-processing

Data pre-processing includes the pre-processing part that we have done before training the architecture. For ResNet 50, 18,101 and Google Net, the pre-processing part is the same. Collected datasets are arranged in two different folders, named them as bacterial spot and healthy. In MATLAB, using the image data store function, we will bring this folder and store it in a variable. After this we split the dataset into training and testing in the form of 75:25, this splitting is done in a random way. Table 1 shows the number of images and class.

Next, we will bring the pretrained model such as ResNet and Google Net. The pre-trained model is trained for 1000 classes, so we change the feature extraction layer and the classification layer of the model. Since we have only two classes for classification, we change number of classes in the feature extraction layer from 1000

**Table 1** Class and image of dataset

Class name	Number of image (approximation)
Bacterial spot	900
Healthy	1400

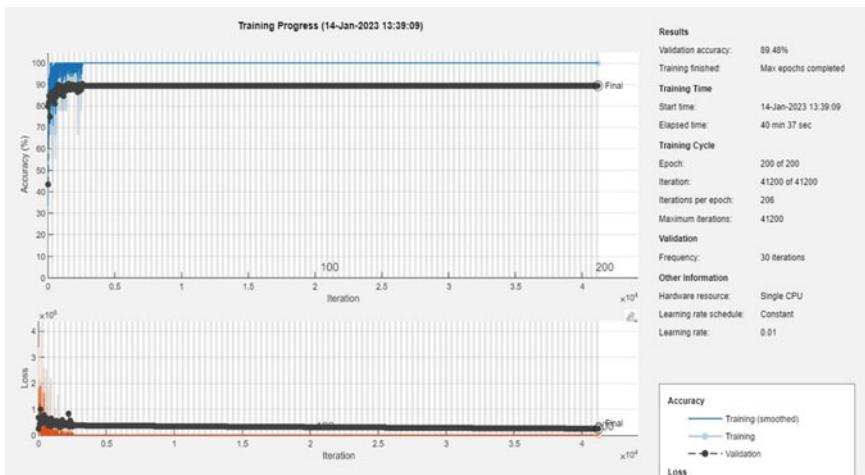
to 2 and output classifier layer in the same way as feature layer. This process is the same for ResNet and Google net.

## Training

For training the pretrained model, we change the hyperparameter for the model. Our aim is to identify the best architecture for disease classification in the bell pepper. Parameter tuning is done in the same manner for ResNet and Google Net. We use stochastic gradient as an optimizer and make mini-batch size as nine and set initial learning rate as 0.01 and validation frequency as 30. Since we are not using the verbose, we change the verbose to false and shuffle every iteration. The training progress of the Google Net is shown in Fig. 8. Table 2 shows the performance of the model that we get after training. Values of Table 2 are approximated.

## Testing

For the testing part, we choose the Google Image of bell pepper leaves that include healthy and bacterial spots and test with different architecture. While we tested with Google Image, we found out that certain images are classified wrongly in all three



**Fig. 8** Training progress of Google Net

**Table 2** Performance of different architectures after training

Architecture	Validation accuracy	Time to complete (min)	Initial validation accuracy	Initial validation loss
ResNet 18	91.10	39 min 38 s	65	$1.2 * 10^6$
ResNet 50	91.26	36 min 51 s	60	$1.2 * 10^6$
ResNet 101	91.10	36 min 51 s	65	$1.2 * 10^6$
Google net	89.48	40 min 37 s	43	$1.2 * 10^6$

cases, these images are not included in the paper. In this paper, we show the resultant image, which are correctly classified by best architectures of convolutional neural network Fig. 9 shows the confusion matrix, confusion chart, performance of Google Net architecture.

Performance of the model is evaluated based on the accuracy, precision, recall, specificity, F1 score.

#### *Accuracy*

Accuracy of the CNN architecture is defined as how much an architecture can predict correctly. It is the total number of correctly predicted to total predictions of the network.

$$\text{Accuracy} = \text{correct predicted} / \text{Total predicted} \quad (1)$$

#### *Precision*

Precision can be defined as the number of predictions that are positively predicted in the total positive prediction. It is important for music, video, e-commerce site

$$\text{Precision} = \text{positively predicted correctly} / \text{Total positive predicted} \quad (2)$$

#### *Recall*

It is defined as the measure of observation in the positive class, which is predicted as positive. It is also referred as sensitive

$$\text{Recall} = \text{Actual positive prediction} / \text{Total actual positive} \quad (3)$$

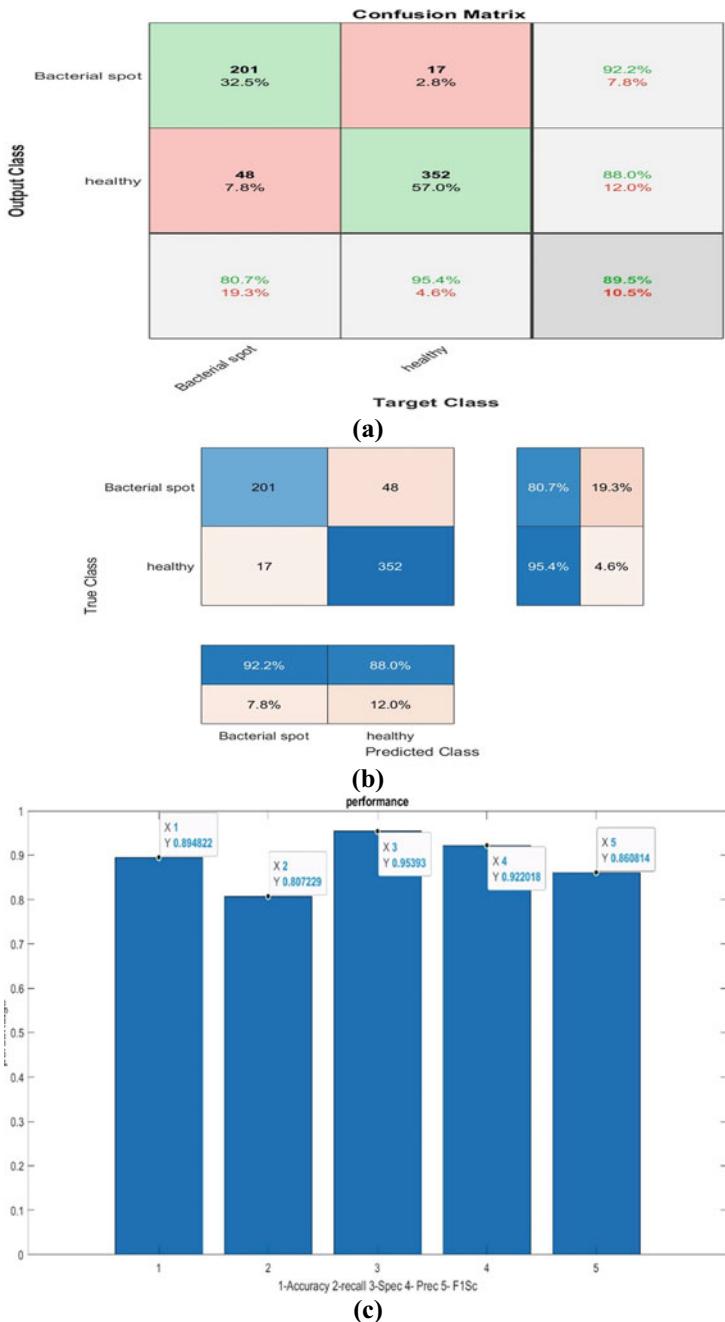
#### *F1 Score*

It is defined as the harmonic mean between precision and recall. It makes a balance between precision and recall.

$$\text{F1 score} = 2 * (\text{recall} * \text{precision}) / \text{recall} + \text{Precision} \quad (4)$$

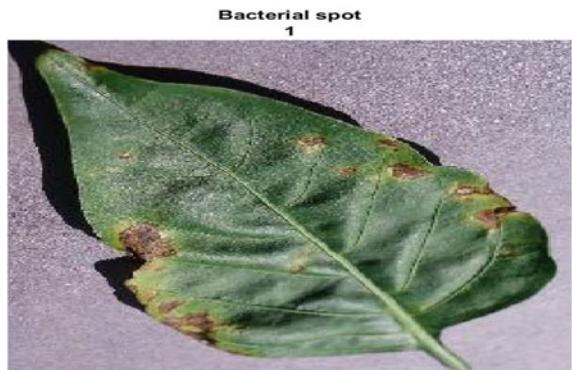
### **5.3 Result Generated**

The result that we obtained after testing images taken from the Google Net is shown in Fig. 10. Since our main purpose of the work is to study the different CNN architectures and compare the capability of the CNN architecture in disease classification, so here we are only including the bacterial spot classification of the image in the bell pepper



**Fig. 9** **a** Confusion matrix, **b** confusion matrix report and **c** performance of Google Net

**Fig. 10** Bacterial spot classification using Google Net



**Table 3** Classification analysis of different architectures

CNN architecture	Classification percentage of bacterial spot	Classification percentage of healthy
Google Net	92.2	88
ResNet 18	87.3	93.9
ResNet 50	89.8	92.7
ResNet 101	87.3	93.9

leaves and also identify the best architecture based on the classification capability of the diseases in bell pepper.

Table 3 shows the classification percentage of different networks on the disease part and healthy part.

From Table 3, we can conclude that disease classification predicted by Google Net is the best compared to other CNN architecture. So, we can conclude that, for bell pepper disease classification, Google Net architecture is more suitable than other architecture. But the drawback is that Google Net faces so many issues such as accuracy, recall, precision, such problems in the disease classification should be addressed in future.

## 6 Conclusion

In India, the majority of people depend on agriculture as their livelihood, so the economy of the country may depend on the production of agricultural yield. But in agriculture, identification of disease at the correct time is a challenging problem. This paper helps the bell pepper farmers to identify the most popular leaf diseases known as bacterial spot using high classification accuracy of CNN architecture known as Google Net. This paper also helps the researchers to analyze different architectures and its performance in a low-cost system. This paper focuses on the importance of

transfer learning in deep learning and also shows a path for the future researchers to analyze these architectures for increasing the performance of the architecture and classification accuracy.

## References

1. Naseem M, Singh V, Ahmed K, Mahroof M, Ahamad G, Abbasi E (2022) Architecture of automatic irrigation system in hilly area using wireless sensor network: a review. In: 2022 2nd international conference on emerging frontiers in electrical and electronic technologies (ICEFEET). Patna, India, pp 1–6. <https://doi.org/10.1109/ICEFEET51821.2022.9847913>
2. Mokhtar U, et al (2015) SVM-based detection of tomato leaves diseases. In: Intelligent Systems' 2014. Springer, pp 641–652
3. Khirade SD, Patil AB (2015) Plant disease detection using image processing. In: 2015 International conference on computing communication control and automation. pp 768–771. <https://doi.org/10.1109/ICCUBEA.2015.153>
4. Sabrol H, Satish K (2016) Tomato plant disease classification in digital images using classification tree. In: Communication and signal processing (ICCP), 2016 international conference on. IEEE. pp 1242–1246
5. Kaya A, Keceli AS, Catal C, Yalic HY, Temucin H, Tekinerdogan B (2019) Analysis of transfer learning for deep neural network-based plant classification models. Comput Electron Agric 158:20–29
6. TM P, Pranathi A, Sai Ashritha K, et al (2018) Tomato leaf disease detection using convolutional neural networks. In: 2018 Eleventh international conference on contemporary computing (IC3)
7. Nachtigall LG, Araujo RM, Nachtigall GR (2016) Classification of apple tree disorders using convolutional neural networks. In: 2016 IEEE 28th international conference on tools with artificial intelligence (ICTAI). pp 472–476. <https://doi.org/10.1109/ICTAI.2016.0078>
8. Rangarajan AK, Purushothaman R, Ramesh A (2018) Tomato crop disease classification using pre-trained deep learning algorithm. In: ELSEVIER, International conference on robotics and smart manufacturing (RoSMA2018)
9. Alzuaidi L, Zhang J, Humaidi AJ, et al (2021) Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 53
10. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 1–9
11. Lin M, Chen Q, Yan S (2013) Network in network. arXiv preprint <arXiv:1312.4400>
12. Bengio Y, et al (2015) Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. arXiv:1502.04390 corr abs/1502.04390
13. Pawara P, Okafor E, Surinta O, Schomaker L, Wiering M (2017) Comparing local descriptors and bags of visual words to deep convolutional neural networks for plant recognition. <https://doi.org/10.5220/0006196204790486>
14. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 770–778
15. Ramzan F, Khan MU, Rehmat A, Iqbal S, Saba T, Rehman A, Mehmood Z (2019) A deep learning approach for automated diagnosis and multi-class classification of Alzheimer's disease stages using resting-state fMRI and residual neural networks. J Med Syst 44 <https://doi.org/10.1007/s10916-019-1475-2>
16. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv preprint <arXiv:1512.03385v1>

# Prediction of Bone Mineral Density Using AI to Detect Osteoporosis



Ritu Chauhan, Yati Varshney, and Harleen Kaur

**Abstract** In this article, we provide a thorough overview of Artificial Intelligence algorithms that may be used for the detection of osteoporosis while measuring bone mineral density with different parameters. This paper discusses cutting-edge technology algorithms of Python such as K-Means, KNN, Linear Bayesian Regression, Naive Bayes Regression, Multi-Linear Regression, Theil Sen Regression, Polynomial fit, Random Forest Classifier, Decision Tree Regression, and certain graphical representations using tableau software on bone mineral density dataset. This data interpretation is entrenched from the bone mineral density dataset which reflects a non-linear coalition of serum uric acid levels concerning the risk of osteoporosis. Also, the study varietally discusses algorithms based on various supervised and unsupervised learning methodologies of machine learning with the help of a bone mineral density dataset.

**Keywords** Machine learning · Python · Tableau · Bone mineral density · Data analysis

## 1 Introduction

Machine learning is an aspect of AI technology that employs computer simulations to assess data and make predictions based on what is learned [1]. It is what gives machines the ability to resolve issues within themselves and produce precise predictions utilizing the available information. Reinforcement learning, unsupervised learning, and supervised learning are only a few of the several types of machine learning [2]. In contrast to unsupervised learning, which utilizes unlabeled data, supervised learning requires training a model upon labeled data. Through trial and

---

R. Chauhan (✉) · Y. Varshney  
Artificial Intelligence and IoT Lab, Center for Computational Biology and Bioinformatics,  
Amity University, Noida, UP, India  
e-mail: [rituchauha@gmail.com](mailto:rituchauha@gmail.com)

H. Kaur  
Department of Computer Science and Engineering, Jamia Hamdard Delhi, New Delhi, India

error, a model is trained through reinforcement learning [3]. Machine learning is a key component of the rapidly expanding field of data science. Ideally, the choices made as a result of these insights affect the key growth metrics of applications and businesses [4]. They will be expected to help identify the most important business questions and the data required to answer them [5]. These algorithms can analyze thousands of images in a matter of seconds and provide accurate and reliable measurements of bone density and structure, which can help healthcare providers diagnose osteoporosis and monitor the progression of the disease. ML algorithms can be used to monitor patients with osteoporosis and track changes in bone density and structure over time. These algorithms can learn to identify patterns and trends in bone density measurements, which can alert healthcare providers to changes that may require further evaluation or intervention.

It is associated with a decrease in skeletal mass in women achieved at age 1, or with a substantially greater loss of BMD with age. This prediction reflects extreme importance to the detection of bone mineral density concerning the weakening of bones [6]. Both energy X-ray absorptiometry is a flexible technique that may assess the density of bones at numerous locations across the body [7].

We have discussed numerous machine learning algorithms and methodology for the bone mineral density dataset which shows a non-linear alliance of serum uric acid levels concerning the risk of osteoporosis. The bone mineral density dataset shows dimensions of clinical evaluation of osteoporosis risk factors and aids in identifying candidates who might benefit from dual-energy X-ray absorptiometry (DEXA), which enables early intervention to effectively lower the incidence of fragility fractures. The goal of this paper is to examine several machine learning approaches based on Python programs, like K-Means, KNN, Linear Bayesian Regression, Naive Bayes Regression, Multi-Linear Regression, Theil Sen Regression, Polynomial fit, Random Forest Classifier, Decision Tree Regression, and certain graphical representations using tableau software on bone mineral density dataset. All the above-mentioned Python programming is done on a Jupyter notebook. The proposed study focused on supervised as well as unsupervised learning techniques of ML algorithms.

## 2 Methodology for Collection of Data

### 2.1 Data Description

We have used different machine learning algorithms using python language. The dataset is collected from the public repository which is known as Kaggle [8]. The bone mineral density contains 1537 rows and 40 columns and has many attributes related to bone mineral density based on gender, age, weight, height, and BMI of the patient. The dataset shows various factors which are responsible for osteoporosis and its co-linear relation with serum uric acid levels.

## 2.2 Machine Learning Algorithms

There are various kinds of algorithms mentioned below based on supervised, unsupervised, and reinforcement learning algorithms.

### K-Means Algorithm

K-means clustering is the simplest unsupervised learning approach used to address clustering challenges [9]. Datasets with the bulk of similarities because datasets are separated into K different clusters based on contrasts and similarities [10].

### K-NN Algorithm

A supervised learning approach called K-Nearest Neighbor can be applied to classification and regression issues [11]. By assuming analogies between both the new information point and the existing data points, this algorithm operates [12].

### Linear Bayesian Regression Algorithm

In Bayesian linear regression, the mean of one parameter is a summation of additional factors [13]. This sort of conditional modeling aims to determine the posterior distribution of both the regressors and other variables affecting the allocation of the regress, and it then permits the prediction of the regress and outside the sample depending on the results of the regression analysis [14]. The Bayesian method has been effectively used and has good mathematical foundations. Therefore, utilizing this does not necessitate any additional dataset expertise [15, 16].

### Naïve Bayes Regression Algorithm

One of the best classifiers that give a decent outcome for a particular situation is the naive Bayes classifier [17]. A Naive Bayesian model is simple to construct and ideal for the vast amount of dataset. It mostly serves to classify texts [18].

### Multi-linear Regression Algorithm

They can be utilized to create a model that successfully forecasts your dependent variable based on the independent factors [19]. This is crucial to remember that before performing the analysis, you should dummy code your dependent variable if it is categorical [20].

### Polynomial Regression Algorithm

The polynomial regression training set has a non-linear structure. It employs a linear regression model to fit the intricate and non-linear functions and datasets [21].

### Theil Sen Regression Algorithm

Theil-Sen evaluation is more resistant to outliers than the OLS (ordinary least squares) estimator. It is advised to utilize Theil-Sen only for modest issues in terms of the number of samples and characteristics due to its computational cost [22].

## Random Forest Classification Algorithm

Random forest classification, i.e., applied for classification and regression issues [23]. This assembly learning strategy that enhances the effectiveness of the model by providing predictions by merging various classifiers.

## Decision Tree Regression Algorithm

An approach for supervised learning called a decision tree is mostly used to tackle classification difficulties, while it can also be used to address regression issues [24]. Both categorical and continuous variables are compatible with it. It displays a tree-like structure with nodes and branches that begin at the root node and grow on additional branches to the leaf node [25].

## 2.3 Tableau

With the help of the model-neutral Tableau Analytics Extensions API, business users can interact with any machine learning model and make decisions in real-time.

## 3 Result

This graph is in Fig. 1 and is intended to demonstrate scenarios in which k-means generate illogical and potentially unwanted groupings.

A 1D regression is shown in Fig. 2 using a decision tree. Thus, it picks up local linear regressions that approximate the sine curve.

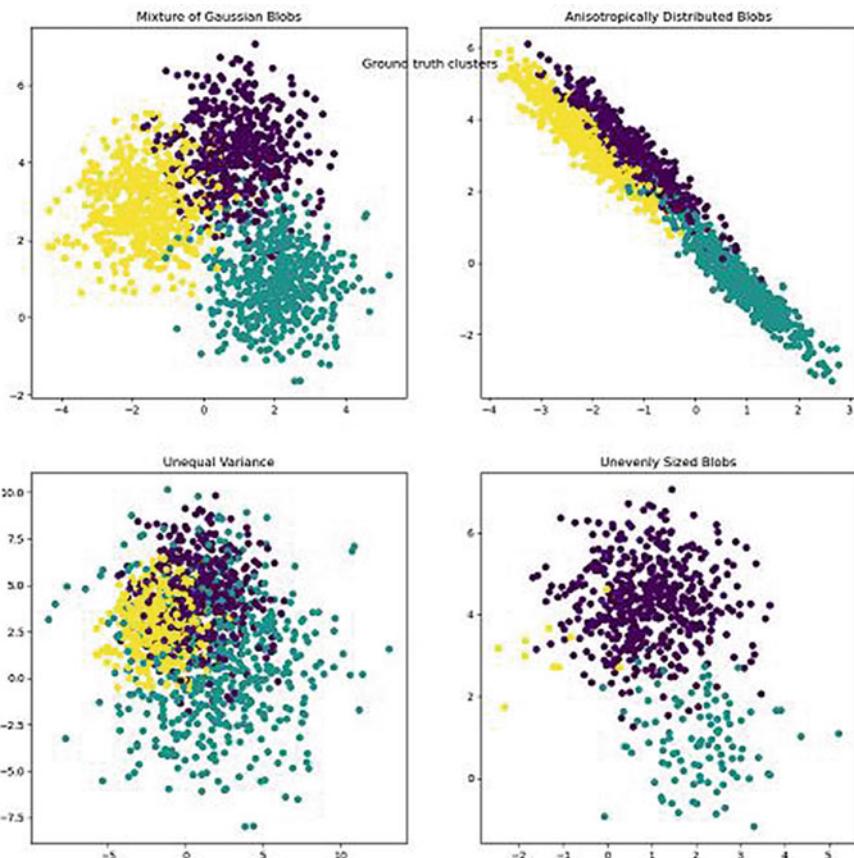
Figure 3, shows how to use a K-Nearest Neighbor to solve a regression problem and how to interpolate a target using both a barycenter and constant weights. Every point in the neighborhood contributes consistently to the categorization of a test point when using the fundamental nearest neighbor's regression.

In Fig. 4, a graph is shown which has models log-likelihood and iteration features. Indeed, up to a flexible cutoff determined by the n\_iter parameter, both models minimize the log-likelihood.

In Fig. 5, the polynomial fit of non-linear features of serum uric acid levels concerning the risk of osteoporosis. The error bars show the query points' expected Gaussian distribution as one standard deviation.

In Fig. 6, Naïve Bayes Regression graph is shown with respect to density and femoral neck (FN) attributes mentioned in the bone mineral density dataset.

In Fig. 7, the graph is showing that smoking is a highly informative feature as compared to others. This Uric acid (URIC), Creatinine (CREA), and Femoral Neck (FN) are moderately informative features among all.



**Fig. 1** K-Means assumption of bone mineral density dataset

**Fig. 2** Decision tree regression using bone mineral density dataset

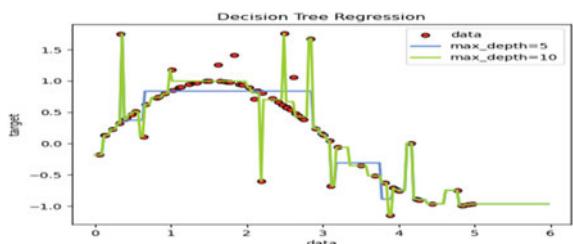
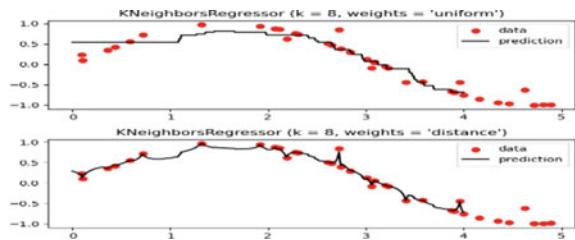


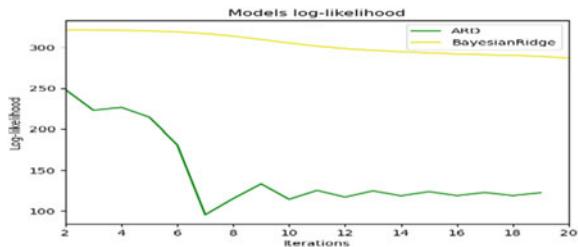
Figure 8, shows the robust approach RANSAC outperforms Theil-Sen. This is demonstrated below, where RANSAC is perturbed by outliers with regard to the x-axis.

In Fig. 9, a multi-linear regression graph is showing actual femoral neck (FN) versus predicted femoral neck features in a multi-linear regression plot graph.

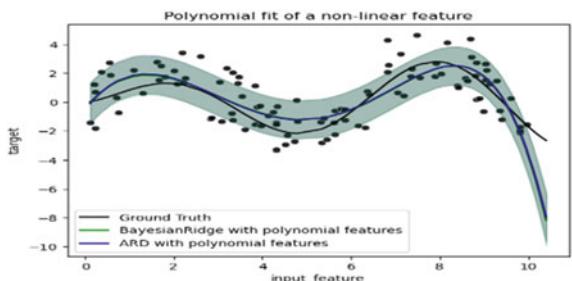
**Fig. 3** K-NN analysis using bone mineral density dataset



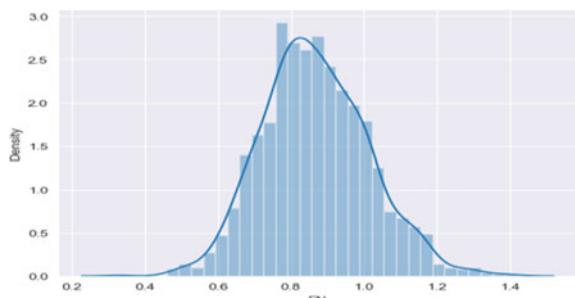
**Fig. 4** Linear Bayesian regressors graph



**Fig. 5** Polynomial fit of non-linear features of bone mineral density

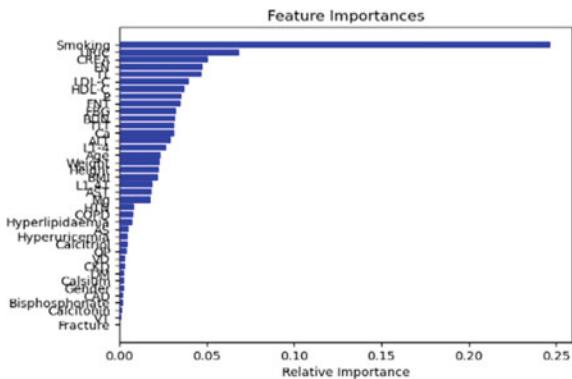


**Fig. 6** Naive Bayes regression plot graphs

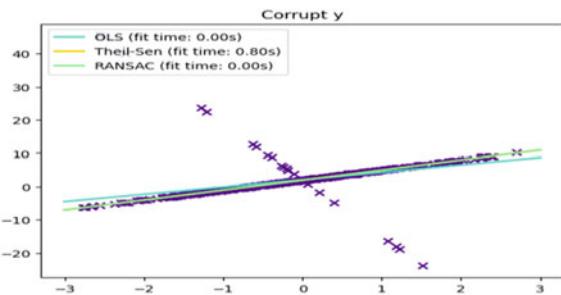


The graph in Fig. 10 is showing the effect of age with respect to the femoral neck and telomere length factors given in the bone mineral density dataset.

In this Fig. 11 with the help of a scatter plot, we can compare HDL-C and LDL-C relation with its effects of drinking and smoking factors.



**Fig. 7** Random forest classifier graph



**Fig. 8** Theil Sen regression of bone mineral density dataset

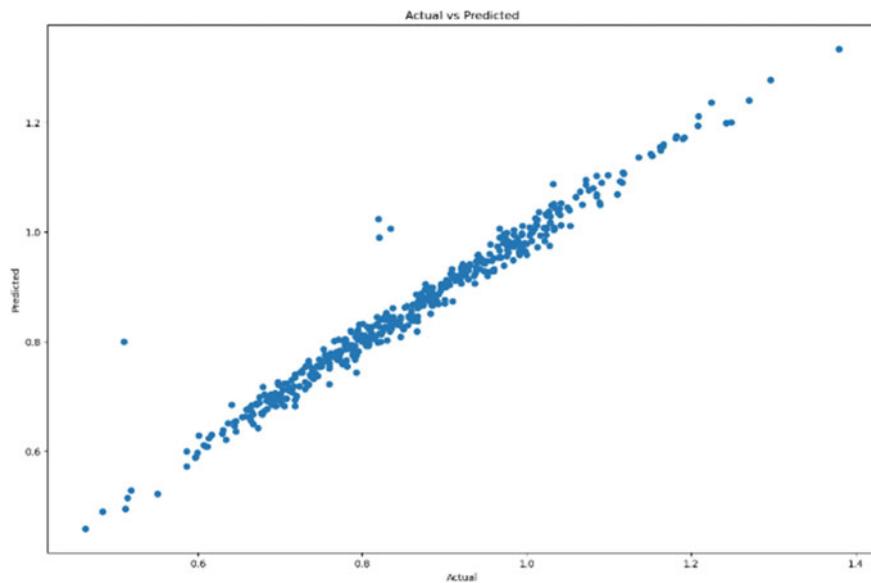
In Fig. 12, graph is created on tableau which reflects Lumbar 1–4 factor with its relation with Alanine Aminotransferase (ALT) and Aspartate Aminotransferase (AST) factors in the given dataset.

In Fig. 13, the graph is created on tableau which reflects the relation of age factor with respect to blood urea nitrogen (BUN). It helps in the early prediction of BUN with respect to different age groups.

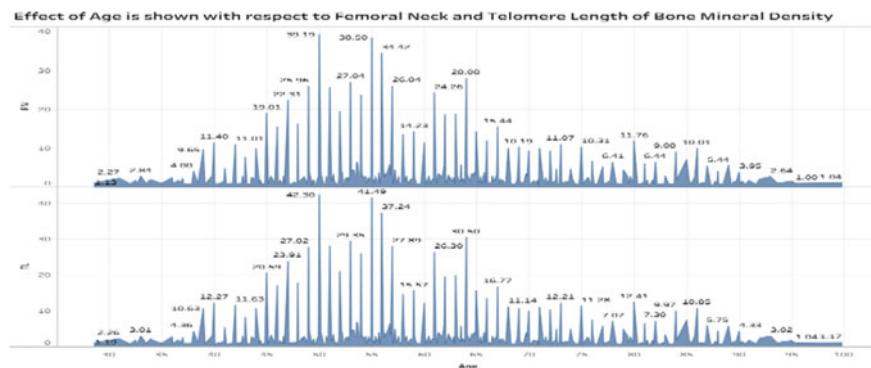
In Fig. 14, the graph is created on tableau which reflects the relation of age factor with respect to fast blood glucose (FBG). It shows how FBG can be a causative factor for early diagnosis of osteoporosis with respect to different age groups.

In Fig. 15, the graphical representation is created by tableau, which shows the relation of Body Mass Index with Hyperlipidemia and Hyperuricemia factors mentioned in the dataset.

In Fig. 16, graph is created with the help of a tableau which shows five factors on a single graph.



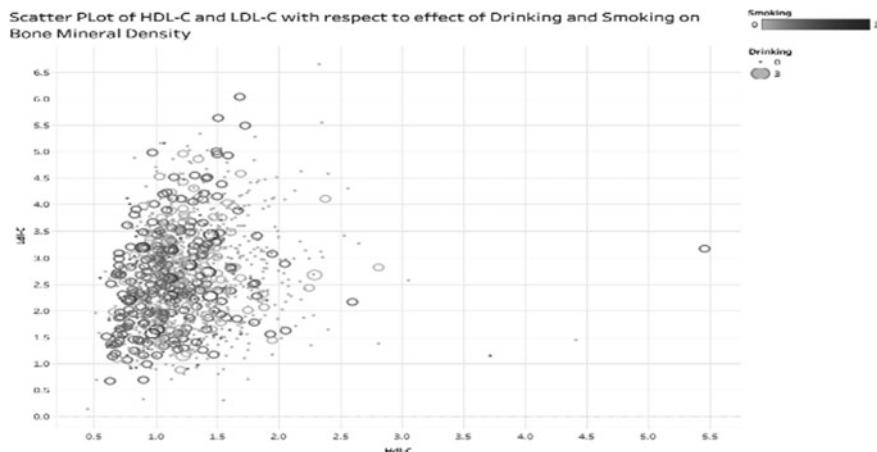
**Fig. 9** Multi-linear regression graph



**Fig. 10** Effect of age is shown with respect to femoral neck and telomere length of bone mineral density

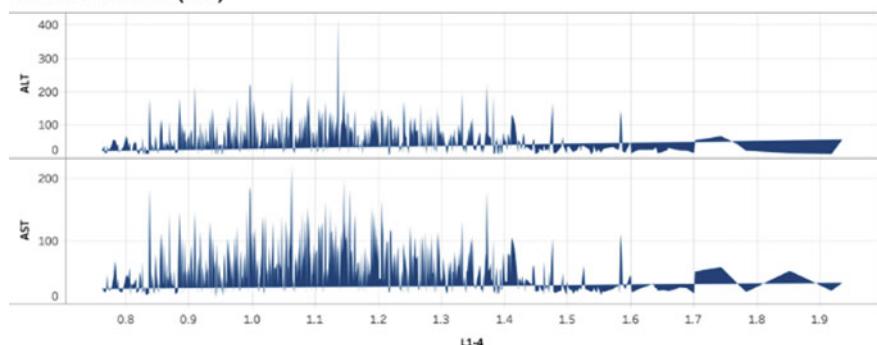
## 4 Conclusions

This paper focuses on machine learning algorithms and tools for the analysis of bone mineral density datasets. There are various python algorithms and graphical representations of the tableau are utilized for the data analysis. The bone mineral density dataset is based on a non-linear coalition of serum uric acid levels concerning the risk of osteoporosis. Furthermore, these techniques can be utilized for many



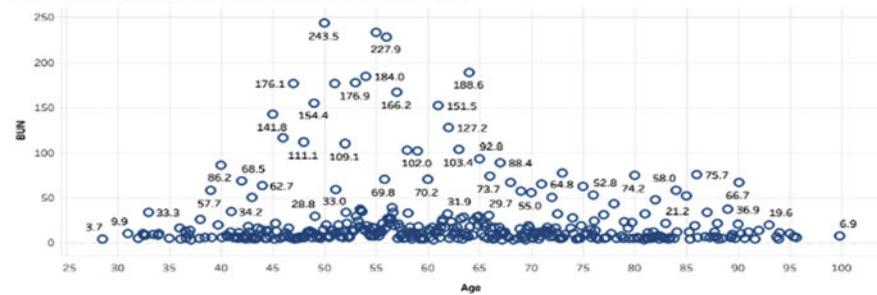
**Fig. 11** Scatter plot of HDL-C and LDL-C

**Lumbar 1-4 bone density with respect to Alanine Aminotransferase (ALT) and Aspartate Aminotransferase (AST)**



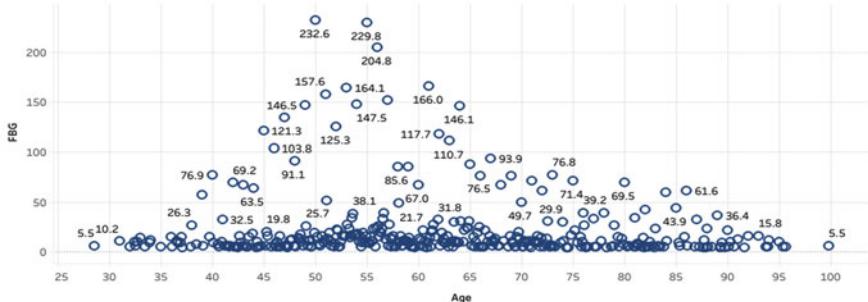
**Fig. 12** L1-4 bone density with respect to ALT and AST

**Relation of Age with respect to Blood Urea Nitrogen**



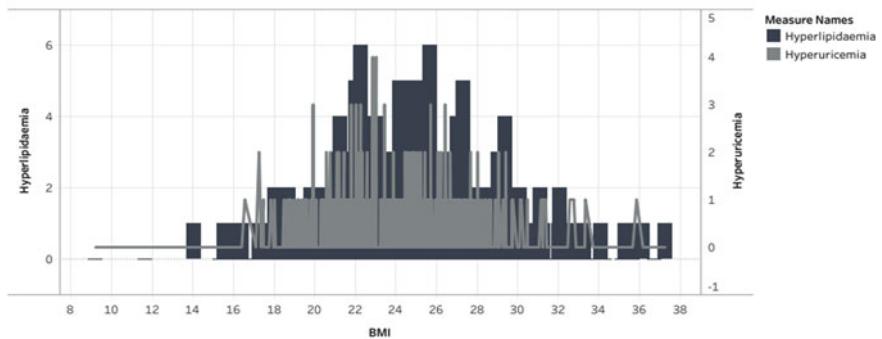
**Fig. 13** Relation of age with respect to BUN

#### Relation of Age with respect to Fast Blood Glucose



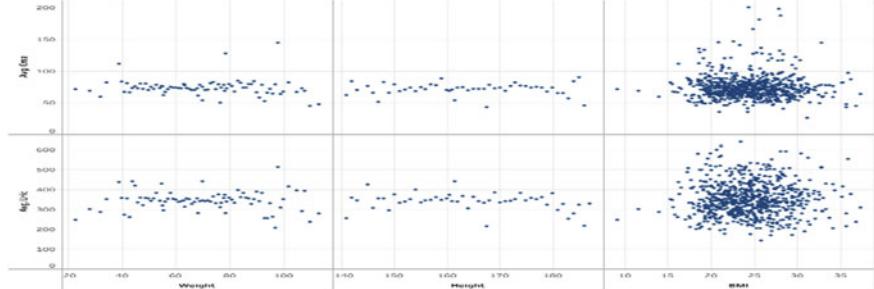
**Fig. 14** Relation of age with respect to fast blood glucose

#### Relation of BMI with respect to Hyperlipidaemia and Hyperuricemia



**Fig. 15** Relation of BMI with respect to Hyperlipidemia and Hyperuricemia

#### Weight, Height and BMI relation with Creatinine and Uric Acid of Bone Mineral Density



**Fig. 16** Weight, height, and BMI relation with creatinine and uric acid of bone mineral density

disease diagnoses and predictions. This paper has shown different machine learning approaches for the analysis of bone mineral density as well as the factors related to bone density.

## References

1. Lima MSM, Delen D (2020) Predicting and explaining corruption across countries: a machine learning approach. *Gov Inf Q* 37(1):101407
2. Bokonda PL, Ouazzani Touhami K, Souissi N (2020) Predictive analysis using machine learning: review of trends and methods. <https://doi.org/10.1109/ISAECT50560.2020.9523703>
3. Castañón J (2019) Machine learning methods that every data scientist should know. Consultado em Outubro 16
4. Devore J (2007) Making sense of data: a practical guide to exploratory data analysis and data mining. *Glenn J Myatt* 67:370–371
5. Pal CJ, Witten IH, Frank E, Hall MA (2016) Data mining: practical machine learning tools and techniques. Morgan Kaufmann, Burlington
6. Sheu A, Diamond T (2016) Bone mineral density: testing for osteoporosis. *Aust Prescr* 39(2):35–9. <https://doi.org/10.18773/austprescr.2016.020>. Epub 2016 Apr 1. PMID: 27340320; PMCID: PMC4917635
7. Arabi A, Baddoura R, Awada H, Khouri N, Haddad S, Ayoub G et al (2007) Discriminative ability of dual-energy X-ray absorptiometry site selection in identifying patients with osteoporotic fractures. *Bone* 40:1060–1065. <https://doi.org/10.1016/j.bone.2006.11.017>
8. Bhathena (2022) Bone mineral density. Bone Mineral Density, Kaggle. <https://datasets/jehanbhathena/bone-mineral-density>. Accessed 15 Jan 2023
9. Wu B (2021) K-means clustering algorithm and Python implementation. In: 2021 IEEE international conference on computer science, artificial intelligence and electronic engineering (CSAIEE), SC, USA, pp 55–59. <https://doi.org/10.1109/CSAIEE54046.2021.9543260>
10. Naeem S (2018) Study and Implementing K-mean clustering algorithm on english text and techniques to find the optimal value of K. *Int J Comp Appl* 182(31):975–8887. <https://doi.org/10.5120/ijca201891823>
11. Guo G, Wang H, Bell D, Bi Y (2004) KNN model-based approach in classification
12. Kubat MJ (2000) Voting nearest-neighbour subclassifiers. In: Proceedings of the 17th international conference on machine learning, ICML-2000. Stanford, CA, pp 503–510
13. Wundervald B (2019) Bayesian linear regression. <https://doi.org/10.13140/RG.2.2.28385.97121>
14. Gelman A, Carlin J, Stern H, Dunson D, Vehtari A, Rubin D (2014) Bayesian data analysis, 3rd edn. CRC, Boca Raton
15. Robert C (2007) The Bayesian choice: from decision-theoretic foundations to computational implementation, 2nd edn. Springer, New York
16. O'Hagan A (1994) Kendall's advanced theory of statistics: Bayesian inference, 2B edn. Arnold, London
17. Yang F-J (2018) An implementation of Naïve Bayes classifier. In: 2018 International conference on computational science and computational intelligence (CSCI). Las Vegas, NV, USA, pp 301–306. <https://doi.org/10.1109/CSCI46756.2018.00065>
18. Rish I (2001) An empirical study of the Naïve Bayes Classifier. In: IJCAI 2001 work empir methods artificial intelligence, vol 3
19. Uyanik GK, Güler N (2013) A study on multiple linear regression analysis. *Procedia Soc Behav Sci* 106:234–240. <https://doi.org/10.1016/j.sbspro.2013.12.027>
20. Trammer M, Elliot M (2008) Multiple linear regression. *Cathie Marsh Centre Cen Surv Res (CCSR)* 5(5):1–5

21. Ostertagová E (2012) Modelling using polynomial regression. Procedia Eng 48:500–506
22. Peng H, Wang S, Wang X (2008) Consistency and asymptotic distribution of the Theil-Sen estimator. J Stat Plan Infer 138(6):1836–1850
23. Mounica M, Vijayasaraswathi R, Vasavi R (2021) Detecting Sybil attack in wireless sensor networks using machine learning algorithms. IOP Conf Ser Mater Sci Eng 1042:012029
24. Vijaya R, Gajavelli K, Nikath A, Vasavi R, Anumasula R (2022) Heart disease prediction using decision tree and SVM. [https://doi.org/10.1007/978-981-16-73894\\_7](https://doi.org/10.1007/978-981-16-73894_7)
25. Patil DV, Bichkar RS (2012) Issues in optimization of decision tree learning

# On the Depth of Convolutional Neural Networks for Image Classification Problems



**Samima Khatun, Md. Yousuf Ansari, Sk. Saidul Alam, and Ayatullah Faruk Mollah**

**Abstract** In this work, we studied the performance changes by increasing the depth of convolutional neural network with four models and four different image datasets. The datasets include MNIST handwritten digits, AUTNT scripts, AUTNT text versus non-text, and Dogs versus Cats. We started with a pair of convolutional layer and max pool layer and gradually expanded the depth by adding pair(s) of convolutional layer and max pool layer. Initially, the accuracy is found to improve with increasing depth up to three pairs for all the four datasets. After that, the accuracy of the models decreases for all datasets. The outcome is, therefore, given using three sets of convolutional and max pool layers. Such study will be useful in designing suitable CNN models for different visual object recognition problems.

**Keywords** CNN · Depth of CNN · Convolutional layer · Max pool layer · AUTNT dataset · MNIST digits dataset

## 1 Introduction

Convolutional Neural Network (CNN) is a baseline of many deep learning architectures and frameworks [1]. It is being successfully applied in a wide range of classification tasks. It is made up of four elemental layers: convolutional, relu, pooling, and fully connected. Its building blocks are filters and kernels that reveal the intrinsic features from the input through convolution. It automatically learns the weights of the filters from the network during training.

These operations are generally expensive. If we use a large number of layers, the model will take more time to train and may overfit. When a model excels at classifying or making predictions on data from the training set but fails miserably to do so on data on which it has not been trained, overfitting takes place. In contrast, if we use fewer layers, the model may underfit. Underfitting is when a model is not

---

S. Khatun · Md. Y. Ansari · Sk. S. Alam · A. F. Mollah (✉)

Department of Computer Science and Engineering, Aliah University, IIA/27 Newtown,  
Kolkata 700160, India

e-mail: [afmollah@aliah.ac.in](mailto:afmollah@aliah.ac.in)

able to properly classify or predict the data on which it is trained. So, the question arises, How many layers should one use to get the maximum accuracy? Researchers often apply trial approach to find a model suitable for them.

Studies related to the impact on CNN performance with resizing and padding [2], image quality enhancement [3], preprocessing [4], activation function, etc. are found in the literature. However, a separate study on how the number of layers affects the performance [5, 6] is limited. In this paper, we studied the impact of increasing the depth of CNN by introducing more and more convolution-pooling pair of layers on a variety of classification problems. Such study will be helpful for designing suitable deep learning models for a given classification problem.

## 2 Related Works and Motivation

In literature, some works on aspects related to the performance of CNN have been reported. For instance, Shahnaz et al. [2] have carried out experiments by applying different resizing and padding techniques on input images and checked the performance of the CNN model. They have considered direct resizing without padding, resizing with zero padding at the bottom/right, resizing with zero padding at the middle, and resizing with estimated background padding. Chan [3], in his Master's thesis, has shown the effect of image enhancement on CNN performance. Pal et al. [4] found that zero component analysis improves CNN performance and it yields better results in image classification than traditional mean normalization and standardization.

Specifically, on the depth of CNN, very few studies have been found. Gorban et al. [5] have presented a study on reducing the number of layers and producing equally good results for pre-trained neural networks. In the work of Seijdel et al. [6], how depth of CNNs may solve in image segmentation is discussed. Josephine et al. [7] experimented with three different deep classification models to analyze and predict early detections of diabetes. The first classification deep model has three dense layers, each with 12, 12, and 8 neurons. The second classification deep model has four dense layers with 32, 24, 12, and 8 neurons each. The final classification model has six dense layers with 64, 64, 32, 24, 12, and 8 neurons in each layer respectively. In this work, from 3 dense layers to 6 dense layers, the classification accuracy increased from 83 to 99%. Finally, it is concluded that the deeper the depth the better the classification performance.

Uzair et al. [8] reviewed different neural networks having different number of hidden layers from different papers. In this paper, they proposed that using three numbers of hidden layers would produce the optimal performance in terms of accuracy and time complexity. The research concluded that as model depth increases, model training becomes slower. Further, this study concluded that the performance of the model depends on the complexity of the data set.

In short, literature reveals that direct studies on how depth of CNN through conv-pool layers influences CNN performance have not been made so far. This may be

helpful to develop insights into the related factors and design suitable CNN models for classification problems. The present work is a modest attempt to fulfill this need. In this work, CNNs of different depths have been experimented on a variety of datasets and comparative study along with analysis have been made to uncover insights.

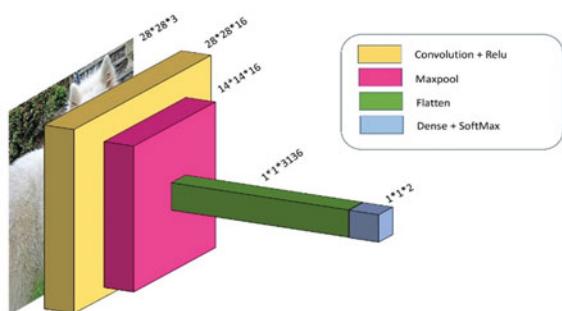
### 3 Network Models

In this work, four different CNN models of increasing depth have been experimented. In each case, the input image data are normalized to  $28 \times 28$  pixels and pixel intensities which typically lie between 0 and 255 are normalized in the range 0.0–1.0. Determining the number of layers in a model is crucial to reveal maximum potential of the network. Selecting a large number of layers may overfit the model and also it is computationally expensive. Conversely, selecting a smaller number of layers may underfit the model. Hence, the number of layers directly affects the performance of the model. To experiment, we started with a pair of convolution and max pool layers and gradually increased the pair. Coming to a solution by comparing a single dataset is inefficient. So, we choose four different datasets to experiment. The detailed description of the models are as follows.

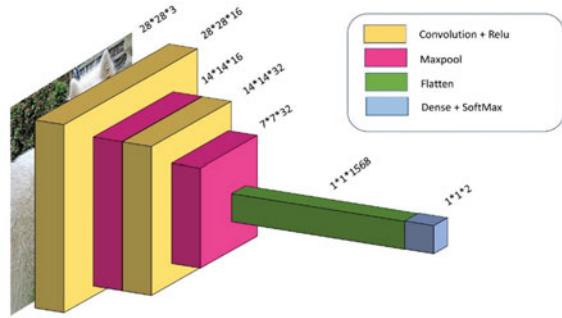
#### 3.1 Depth Level 1 (Model 1)

The first model, as shown in Fig. 1, forms with one convolution layer, with 16 number of  $3 \times 3$  filters with relu activation function followed by one max pool layer with filter size  $2 \times 2$  and stride size 2, which is used for dimensionality reduction by taking max values in a region. Then, we added one flatten layer to transform matrix into 1D vector. Finally, we add output dense layer with a ‘softmax’ activation function for probability output.

**Fig. 1** Depth level 1 architecture



**Fig. 2** Depth level 2 architecture



### 3.2 Depth Level 2 (Model 2)

The second model forms with two conv-pool pairs as shown in Fig. 2. The first convolution layer contains  $16 3 \times 3$  filters with relu activation function followed by one max pool layer with filter size  $2 \times 2$  and stride size 2. The second convolution layer contains  $32 3 \times 3$  filters and relu activation function followed by one max pool layer with filter size  $2 \times 2$  for stride size 2. Then, flatten layer is added to transform the feature matrices into 1D vector. Finally, we add output dense layer with a ‘softmax’ activation function for probability output.

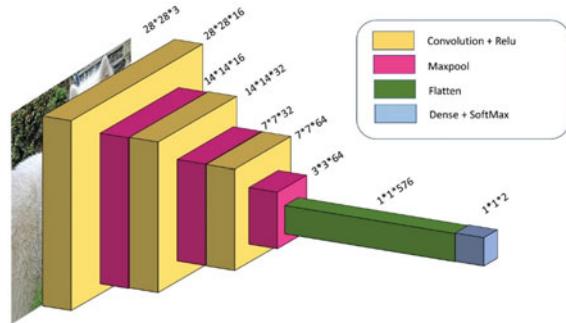
### 3.3 Depth Level 3 (Model 3)

As shown in Fig. 3, the third model forms with three conv-pool pairs. The first one contains  $16 3 \times 3$  filters and relu activation function followed by one max pool layer with filter size  $2 \times 2$  for stride size of 2. The second one contains  $32 3 \times 3$  filters and relu activation function followed by one max pool layer with filter size  $2 \times 2$  strides 2. The third one contains  $64 3 \times 3$  filters and relu activation function followed by one max pool layer with filter size  $2 \times 2$  strides 2. Then, a flatten layer is added to transform the feature maps into 1D vector. Finally, we add dense layer with a ‘softmax’ activation function for probability output.

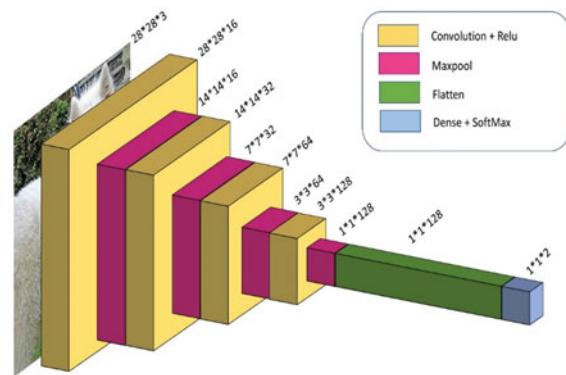
### 3.4 Depth Level 4 (Model 4)

The fourth model forms with four conv-pool pairs. The first three pairs are similar to that of model 3 as evident in Fig. 4. The fourth one contains  $128 3 \times 3$  filters and relu activation function followed by one max pool layer with filter size  $2 \times 2$  strides 2. The flatten layer and dense layers are alike with the model 3.

**Fig. 3** Depth level 3 architecture



**Fig. 4** Depth level 4 architecture



## 4 Experiments and Observation

The implementation was based on Tensorflow 2.0 and Keras 2.3.1 APIs on Python 3.7.11 platform. Brief description of the datasets considered in the experiments is included in Sect. 4.1, the performance of different CNN models has been shown in Sect. 4.2 and in Sect. 4.3 we present the findings and related discussion. It may be noted that we divided the dataset into training, validation and test sets. In all experiments, we have applied Adam optimizer and categorical cross entropy loss function, and training is conducted for 10 epochs only due to computational constraints.

### 4.1 Datasets Description

Four different sets of data have been used for carrying out experiments viz. Modified National Institute of Standards and Technology (MNIST) digits [9], AUTNT scripts [10], AUTNT text versus non-text [11], and Kaggle cats versus dogs [12]. Summary of these datasets is outlined in Table 1. Sample images from each of these datasets are shown in Fig. 5.

**Table 1** Brief particulars of the datasets considered in evaluating the performance of CNNs of increasing depth

Dataset	Particulars	#Class	#Training samples	#Test samples
MNIST [9]	English handwritten digits images (0–9)	10	60,000	10,000
AUTNT scripts [10]	Text component images of Bengali, Devanagari and Latin script	3	6,314	1,576
AUTNT text versus non-text [11]	Images of text and non-text components	2	8,619	2,152
Dogs versus cats [12]	Images of dogs and cats	2	5,000	1,000

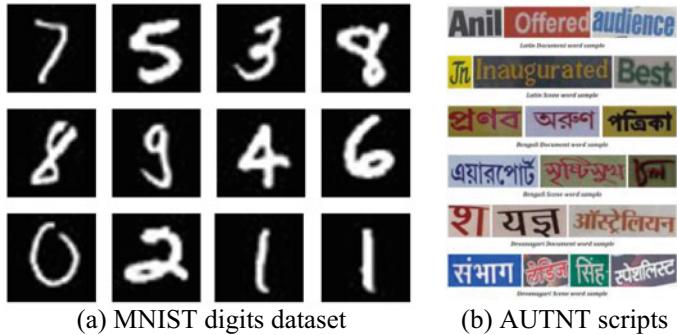
## 4.2 Prediction Performance

We analyzed the performance of the models by comparing confusion matrices with varying depth levels. In the confusion matrix of the test set for MNIST handwritten digit dataset [9], we can see that out of 10,000 samples the model for depth level 1 has misclassified 168 times, depth level 2 has misclassified 98 times, depth level 3 has misclassified 83 times and depth level 4 has misclassified 127 times. We can observe that misclassification from depth level 1 to 2 decreases by 70, from depth level 2 to 3 decreases by 15 but from depth level 3 to 4 increases by 44. Clearly, the model for depth level 3 performs the best.

The AUTNT Scripts dataset [10] contains images of multi-script text components, i.e., Bengali, Devanagari and Latin. Here indices are of these three classes. We have a total of 1576 images to test the models out of which 502 are Bengali, 321 are Devanagari and 753 are Latin. From the confusion matrix we can see that out of 1576 samples, the model for depth level 1 is misclassified 178 times, for depth level 2 is misclassified 151 times, for depth level 3 is misclassified 122 times and for depth level 4 is misclassified 149 times. We can observe that misclassification from depth level 1 to 2 decreases by 27, from depth level 2 to 3 decreases by 29 but from depth level 3 to 4 increases by 27. Clearly for depth level 3 model performs the best.

For the AUTNT text versus non-text dataset [11], we have a total of 2152 images to test the models out of which 1576 are text script and 576 are non-text images. From the confusion matrix we can see that out of 2152 samples, the model for depth level 1 is misclassified 56 times, for depth level 2 is misclassified 46 times, for depth level 3 is misclassified 41 times and for depth level 4 is misclassified 63 times. We can observe that misclassification from depth level 1 to 2 decreases by 10, from depth level 2 to 3 decreases by 5 but from depth level 3 to 4 increases by 22. Clearly for depth level 3 model perform best.

The Dogs versus Cats dataset [12] contains images of dogs and cats. To test the models, we took 1000 images of dogs and cats, including 500 dog and 500 cat images. From the confusion matrix we can see that out of 1000 samples, the model for depth



(c) AUTNT text vs non-text



(d) Dogs vs. Cats

**Fig. 5** Sample images from the four sets of data (MNIST [9], AUTNT scripts [10]. AUTNT text versus non-text [11], dogs versus cats [12])

level 1 is misclassified 285 times, for depth level 2 is misclassified 259 times, for depth level 3 is misclassified 244 times and for depth level 4 is misclassified 272 times. We can observe that misclassification from depth level 1 to 2 decreases by 26, from depth level 2 to 3 decreases by 15 but from depth level 3 to 4 increases by 28. Again, for depth level 3 model performed the best.

From the confusion matrices, we observed that with one set of convolution and max pool layer (depth level 1), misclassification is highest for all four datasets. The misclassification rate decreases for all four datasets when adding another set of convolution and max pool layers (depth level 2). Again, by adding one more set of convolution and max pool layer (depth level 3), the misclassification rate decreases for all four datasets. However, adding one more set of convolution and maximum pool layers increases the misclassification rate for all four datasets. We can conclude that the model with depth level 3 performs the best.

### 4.3 Comparative Study and Discussion

Table 2 shows the model performance of all four datasets. It can be seen that the highest accuracy is 99.17% for the MNIST dataset, 92.26% for the AUTNT scripts dataset, 98.09% for AUTNT text versus non-text, and 75.60% for the dog versus cat dataset, which results from Model 3. From Table 2, we can find a pattern that by increasing the depth level from 1 to 2 and then 3, the accuracy of the model increased, but after another depth increase, the accuracy of the model decreased.

**Table 2** Prediction performance of CNNs of increasing depth levels for four sets of different data

Dataset	Model	Precision	Recall	F-M	ACC	AUC	RMSE
MNIST (Digits)	Model 1	0.9833	0.9832	0.9832	0.9832	0.9997	0.5686
	Model 2	0.9902	0.9902	0.9902	0.9902	0.9999	0.4274
	Model 3	0.9917	0.9917	0.9917	<b>0.9917</b>	0.9999	0.3576
	Model 4	0.9874	0.9873	0.9873	0.9873	0.9998	0.4678
AUTNT (Scripts)	Model 1	0.8893	0.8871	0.8870	0.8871	0.9648	0.4624
	Model 2	0.9056	0.9042	0.9018	0.9042	0.9765	0.4435
	Model 3	0.9274	0.9226	0.9235	<b>0.9226</b>	0.9835	0.4245
	Model 4	0.9198	0.9055	0.9059	0.9055	0.9864	0.4527
AUTNT (text versus non-text)	Model 1	0.9749	0.9740	0.9742	0.9740	0.9756	0.1613
	Model 2	0.9791	0.9786	0.9787	0.9786	0.9787	0.1462
	Model 3	0.9811	0.9809	0.9810	<b>0.9809</b>	0.9787	0.1380
	Model 4	0.9706	0.9707	0.9706	0.9707	0.9579	0.1710
Dogs versus cats	Model 1	0.7150	0.7150	0.7150	0.7150	0.7149	0.5338
	Model 2	0.7428	0.7410	0.7405	0.7410	0.7410	0.5089
	Model 3	0.7561	0.7560	0.7560	<b>0.7560</b>	0.7560	0.4939
	Model 4	0.7323	0.7280	0.7267	0.7280	0.7280	0.5215

## 5 Conclusion

From our study, we can conclude that increasing the depth to a certain level of the model increases the accuracy. For these small types of datasets, three pairs of convolution and max pool layers perform the best. With four pairs of layers, the model is overfitting and with less than three layers, the model is underfitting. Finding the optimal depth is still challenging. However, this empirical study would cater to understanding about the impact of increasing conv-pool layers on overall prediction performance of CNN. Such understanding, in turn, will foster insights and be helpful for designing suitable CNN models for different image classification tasks. In future, we can expand our study to include large datasets. We may further expand or reduce the depth. It may be interesting to see how the model performs.

## References

1. Khan T, Sarkar R, Mollah AF (2021) Deep learning approaches to scene text detection: a comprehensive review. *Artif Intell Rev* 54(5):3239–3298
2. Shahnaz M, Mollah AF (2022) On the performance of convolutional neural networks with resizing and padding, In: Proceedings of 2nd international conference on frontiers in computing and systems. Springer, pp 51–62
3. Chan X (2019) Image enhancement effect on the performance of convolutional neural network. Master Thesis at Blekinge Institute of Technology, Sweden
4. Pal KK, Sudeep KS (2016) Preprocessing for image classification by convolutional neural networks. In: IEEE International conference on recent trends in electronics, information & communication technology, pp 1778–1781
5. Gorban AN, Mirkes EM, Tyukin IY (2020) How deep should be the depth of convolutional neural networks: a backyard dog case study. *Cogn Comput* 12(2):388–397
6. Seijdel N, Tsakmakidis N, De-Haan EH, Bohte SM, Scholte HS (2020) Depth in convolutional neural networks solves scene segmentation. *PLoS Comput Biol* 16(7):e1008022
7. Josephine VH, Nirmala AP, Alluri VL (2021) Impact of hidden dense layers in convolutional neural network to enhance performance of classification model. *IOP Conf Ser Mater Sci Eng* 1131(1):012007
8. Uzair M, Jamil N (2020) Effects of hidden layers on the efficiency of neural networks. In: Proceedings of 23rd international multtopic conference. IEEE, pp 1–6
9. The MNIST database of handwritten digits. <http://yann.lecun.com/exdb/mnist>. Accessed 08 June 2022
10. Khan T, Mollah AF (2021) Component-level script classification benchmark with CNN on AUTNT dataset. In: Proceedings of international conference on frontiers in computing and systems. Springer, pp 225–234
11. Khan T, Mollah AF (2019) AUTNT - A component level dataset for text non-text classification and benchmarking with novel script invariant feature descriptors and D-CNN. *Multimedia Tools Appl* 78(22):32159–32186
12. Dogs versus cats dataset. [www.kaggle.com/c/dogs-vs-cats](http://www.kaggle.com/c/dogs-vs-cats). Accessed 08 June 2022

# A Modified Tilt Integral Derivative (MTID) Controller Design Using Direct Synthesis Approach



Rammurti Meena, Vipin Chandra Pal, and Sudipta Chakraborty

**Abstract** In this study, direct synthesis (DS) technique for designing non-integer order (NIO) controllers for NIO processes is presented. The most prevalent method for setting the proportional-integral-derivative (PID) controllers is the DS approach. However, NIO processes have not yet been subjected to this methodology. The DS approach is used in this study to take advantage of its simplicity. The resulting controllers are like the enhanced tilt-integral-derivative (TID) controller. To figure out the effectiveness and robust analysis of the proposed control law, simulation studies on DC servo systems and Bioreactors are performed and compared with state of art control laws. Simulations are done for the nominal models as well as perturbed process models. The analysis of robustness of the suggested control law is done by employing 10 percent variation in plant parameters. Lastly, a performance evaluation of different errors is also included.

**Keywords** Direct synthesis (DS) · Tilt-integral-derivative (TID) · PID control · Non-integer order (NIO) · Time-delay

## 1 Introduction

The most common type of controller for industrial operations is the PID controller by Chakraborty et al. [1, 2], Das et al. [3] and Sengupta et al. [4]. Fractional calculus basically deals with integrators and derivatives of fractional order. Because of the intrinsic characteristics of many industrial processes, fractional differential equations can be used to model them more effectively. The growing application of the same in control engineering has caught the attention of scientific communities. The application of fractional calculus is to spread into a select few fields, including electrochemistry, generalized voltage dividers, control theory, and models of neurons in biology by Debnath [5] and Sun et al. [6]. Furthermore, fractional controllers are

---

R. Meena (✉) · V. C. Pal · S. Chakraborty

Electronics and Instrumentation Engineering Department, National Institute of Technology Silchar, Assam 788010, India

e-mail: rammurti21\_rs@ei.nits.ac.in

well known for offering greater design flexibility and durability because of a greater number of tuning options by Kumar et al. [7], Zheng et al. [8] and Bettayeb et al. [9]. For that reason, this area of research has been identified by many researchers in recent times and some recent works are cited below.

## **1.1 DOF Control Schemes**

Using frequency domain specifications an integral with proportional plus derivative control scheme is presented by Chakraborty et al. [10]. A controller design for two-PD gains based on stability margins is developed by Chakraborty et al. [11]. Using stability criterion, a new structure for integrating plants has been developed by Das et al. [12]. An upgraded smith-predictor based control law for NIO process is proposed by Meena et al. [13]. A PSO based fractional PID control law is suggested by Doostinia et al. [14]. A comparative robustness study among Skogestad internal model control, Fractional order PID (FOPID), and conventional PID is shown by Viola et al. [15]. Saxena et al. [16] have given a PI control law having a fractional integrator using IMC framework and Bode's ideal characteristics. For stable, unstable, and integrating processes, Arya and Chakrabarty [17] provide a PID control with a filter with gain and phase margins on an improved IMC paradigm.

## **1.2 DOF Control Schemes on Fractional Order Framework**

To maintain the tradeoff between servo and regulatory action, many researchers have developed different types of dual degree-of-freedom (dual DoF) control laws for both integer and NIO plants. A dual DoF NIO fuzzy PID control law using sorting genetic algorithm-II is presented by Mohan et al. [18]. Shalaby et al. [19] developed a machine learning-based online tuning approach for FOPIID to tackle the adverse effects of parameter fluctuations and disruptions. An optimization based dual-loop fractional order PID control law to control PMSM drives is designed by Jakovljević et al. [20]. A comparison-based study of dual DoF PID and dual DOF NIOPID controllers is studied by Bingi et al. [21]. Based on CSA and SGA-II, a dual DoF FOPIID control scheme is designed by Pachauri et al. [22]. Using IMC scheme, a two DoF controller for 2nd-order unbounded systems is proposed by Kumari et al. [23].

The paper is organized as: In the first section, fractional calculus introduction and a literature review of 1DOF and 2DOF control systems are presented. Methodology of controller design is discussed in Sect. 2. In Sect. 3, simulation outcomes and case study discussions on various industrial process models, such as DC motors and bioreactors, are provided. In Sect. 4, there is a final observation made regarding the proposed work.

## 2 Methodology

### 2.1 Direct Synthesis Method

Several schemes for controller design were reported by many researchers to get the desired tracking response and load-disturbance rejection without affecting robustness of the closed loop response. The most prevalent method for fine-tuning of PID controllers is the direct synthesis (DS) approach. Nevertheless, fractional order models have not utilized this strategy. In the DS technique, the desired control law is achieved by selecting a desired transfer function of closed loop (CLTF), using an approximated open loop process model. Let us consider a conventional feedback control system with unit feedback as depicted in Fig. 1. Here,  $G_p(s)$  is the plant with controller  $C(s)$ . Here,  $R(s)$  is the reference input,  $E(s)$  denotes the difference between  $Y(s)$  and  $R(s)$ ,  $D(s)$  is the external disturbance,  $U(s)$  and  $Y(s)$  denote control action and output, respectively.

From Fig. 1 the CLTF can be defined as:

$$G_{cl} = \frac{G_p(s)C(s)}{1 + G_p(s)C(s)} \quad (1)$$

To get the desired CL response, the controller can be calculated from Eq. (1) as:

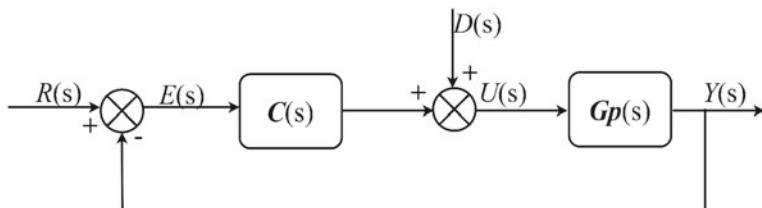
$$C(s) = \frac{1}{G_p(s)} \left( \frac{G_{cl}(s)}{1 - G_{cl}(s)} \right) \quad (2)$$

Integer and fractional order models are both compatible with the controller expression (2).

Explicit tuning formulas for a few well-researched plant models are said to exist. The following plant models have been considered:

Process 1:

$$G_p(s) = \frac{K}{\alpha s^q + 1} e^{-Ls} \quad (3)$$



**Fig. 1** Conventional unity feedback control structure

Process 2:

$$G_p(s) = \frac{K}{\alpha_2 s^{q_2} + \alpha_1 s^{q_1} + 1} e^{-Ls} \quad (4)$$

where  $K$  represents the gain of open loop,  $\alpha$ ,  $\alpha_1$  &  $\alpha_2$  are the time constants,  $q$ ,  $q_1$  &  $q_2$  are integer order or fractional orders and  $L$  is the dead time term of the process.

## 2.2 Controller Design for Processes 1

Assuming the desired CLTF as:

$$G_{cl} = \frac{1}{\alpha_{cl}s + 1} e^{-Ls} \quad (5)$$

Now, using the DS method, a TI control is developed by utilizing 1<sup>st</sup> order Taylor series approximation ( $e^{-Ls} \approx 1 - Ls$ ) to estimate the dead time term in the denominator of Eq. (2):

$$C(s) = \frac{1}{K} \frac{\alpha}{\alpha_{cl} + L} \left( s^{q-1} + \frac{1}{\alpha s} \right) \quad (6)$$

The TF in (6) demonstrates that the controller can take different forms depending on the value of  $q$ . For  $q < 1$ , the controller in (6) includes a fractional integral term in addition to a normal (order 1) integral term, and the structure is referred to as the tilted or shaped structure. If this structure has an additional derivative term, the controller is known as the T ID controller. The suggested approach yields a control law with a conventional integral term with a NI derivative term for  $q > 1$ . One obtains a P I control law for  $q = 1$ .

## 2.3 Controller Design for Processes 2

Assuming the desired CLTF as:

$$G_{cl} = \frac{1}{\alpha_{cl}s + 1} e^{-Ls} \quad (7)$$

The controller can be obtained as:

$$C(s) = \frac{1}{K(\alpha_{cl} + L)} \left( \alpha_2 s^{q_2-1} + \alpha_1 s^{q_1-1} + \frac{1}{s} \right) \quad (8)$$

Typically processes as in Eq. (4) are chosen with  $q_1 \leq 1 < q_2$ . So, the final controller consists of a shaped element having order  $1 - q_1$ , along with an integral and a fractional derivative term.

## 2.4 Controller Structure

The controllers in (6) and (8) are having similar structures to TID control having a NI derivative element. This is referred to as a  $T^v ID$  controller and define the general form of transfer function as:

$$C(s) = Ts^{-v} + \frac{I}{s} + Ds^\mu \quad (9)$$

Using the above notations, the required closed loop response of forms (6 & 8) and Table 1 summarized the controller parameters for two different types of processes.

## 2.5 Selection of Controller Parameter $\alpha_{cl}$

The only need for the DS approach is to provide the appropriate closed loop transfer function. A first order unit gain response was considered with the same open loop delay for the formulations as the desired response. A certain fraction of  $\alpha$  can be chosen as  $\alpha_{cl}$ . A value of  $\alpha_{cl}$  between 1 and 4 is chosen for the simulation results. For processes 2 in Eq. (4)  $\alpha = \sqrt{\alpha_2}$  is used.

## 3 Results and Discussions

The following simulation studies are conducted to prove the productiveness and robustness of the suggested approach and compared with state of artwork. A comparison of performance based on different errors like ISE, ITAE, and IAE is also

**Table 1** Parameter tuning for different types of process models

Process model	$T$	$v$	$I$	$D$	$\mu$
$\frac{K}{\alpha s^q + 1} e^{-Ls}, q \leq 1$	$\frac{1}{K} \frac{\alpha}{\alpha_{cl} + L}$	$1 - q$	$\frac{1}{K} \frac{1}{\alpha_{cl} + L}$	0	-
$\frac{K}{\alpha s^q + 1} e^{-Ls}, q > 1$	0	-	$\frac{1}{K} \frac{1}{\alpha_{cl} + L}$	$\frac{1}{K} \frac{\alpha}{\alpha_{cl} + L}$	$q - 1$
$\frac{K}{\alpha_2 s^{q_2} + \alpha_1 s^{q_1} + 1} e^{-Ls}, q_1 \leq 1 < q_2$	$\frac{1}{K} \frac{\alpha_1}{\alpha_{cl} + L}$	$1 - q_1$	$\frac{1}{K} \frac{1}{\alpha_{cl} + L}$	$\frac{1}{K} \frac{\alpha_2}{\alpha_{cl} + L}$	$q_2 - 1$

included. Where,

$$ISE = \int E^2 dt, ITAE = \int t|E|dt, IAE = \int |E|dt$$

### 3.1 Case Study-1

One of the key elements of machinery in the process sector, particularly for the control aspect, is the DC servo motor. Let us use a model of a DC servo system from Arya and Chakraborty [24] as:

$$G_1(s) = \frac{1643.4}{0.071s^{0.967} + 1} e^{-0.0547s} \quad (10)$$

Using  $\frac{\alpha}{\alpha_{cl}} = 4$ , the proposed controller settings are obtained as:

$$[T \ v \ I] = [0.000596 \ 0.033 \ 0.008399] \quad (11)$$

According to Arya and Chakraborty's [24] design, the FO-IMC controller is:

$$C(s) = \left[ \frac{0.071s^{0.967} + 1}{1643.4(0.0559s^{1.083} + 0.0547s)} \right] \quad (12)$$

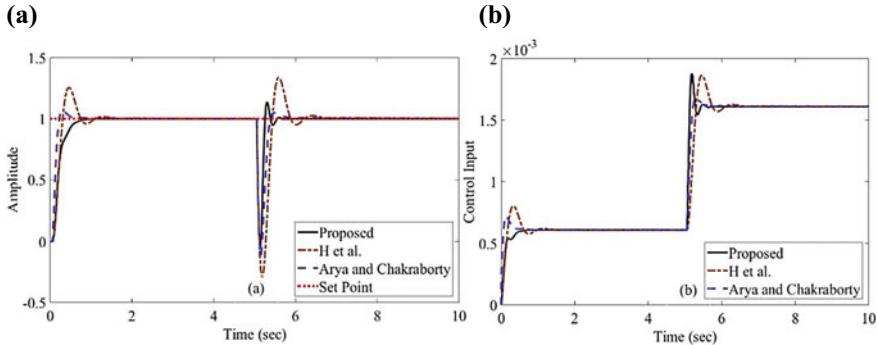
For the design of H et al. [25], the FO-PI controller settings are as follows:

$$C_{FOPI}(s) = 6.1681 \times 10^{-7} \left[ 1 + \frac{7139}{s^{1.053}} \right] \quad (13)$$

The nominal response of the plant is included in Fig. 2a and control actions corresponding to Fig. 2a are shown in Fig. 2b. Simulations are carried out by taking  $R(s) = 1$  at  $t = 0$  second and an external disturbance  $D(s) = -0.001$  at  $t = 5$  second. Table 2 displays a performance comparison for this study based on various errors. To demonstrate the viability of the suggested control law, simulations with 10% change in plant gain and 10% change in plant delay are conducted, and the corresponding outputs are given in Fig. 3a, b.

#### Case study-1: observation

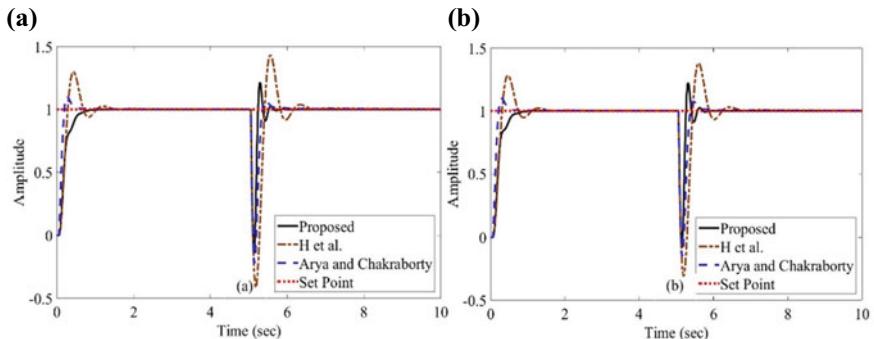
In comparison to existing controller designs, the results show that the presented technique offers better servo tracking and regulatory action. Comparative performance on errors is also seen to be superior. It is clear from Fig. 3a, b that the process uncertainty causes only slight variations in the response of closed loop for the given one.



**Fig. 2** (a) Nominal response of plant in  $G_1(s)$  (b) Control inputs

**Table 2** ISE, ITAE, and IAE comparison of control performance for case study 1

Method	ISE	ITAE	IAE
Presented	0.1183	0.7511	0.215
Arya and Chakrabarty [24]	0.2615	1.22	0.3708
H et al. [25]	0.4976	2.404	0.7075



**Fig. 3** Perturbed response of  $G_1(s)$  (a) 10% changes in gain (b) 10% changes in delay

### 3.2 Case Study-2

The bioreactor is a continuously running process with a continuous flow of both input and output. A continuous operation is chosen for high volume production because it is simple to operate. Let us use the approximated bioreactor model from Trivedi and Padhy [26] as follows:

$$G_2(s) = \frac{5.8}{50.3s^{1.12} + 7.8s^{0.95} + 1} e^{-1.23s} \quad (14)$$

Using  $\frac{\alpha}{\alpha_{cl}} = 4$ , the proposed controller settings are obtained as:

$$[T \ v \ I \ D \ \mu] = [0.1616 \ 0.05 \ 0.0207 \ 1.042 \ 0.12] \quad (15)$$

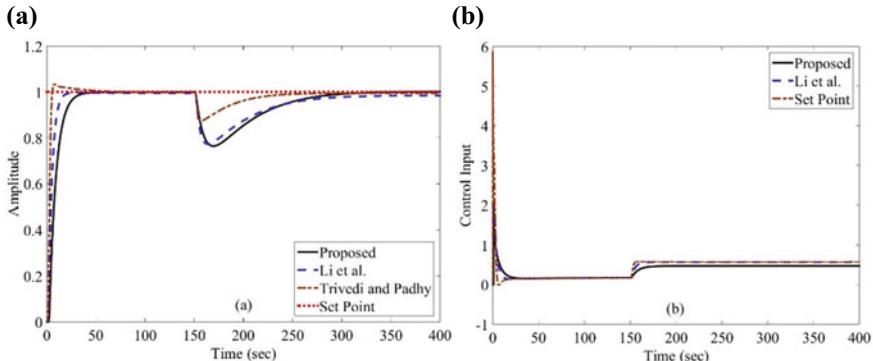
According to Trivedi and Padhy's [26] design, the controller settings are as follows:

$$C(s) = 0.6217 \left[ 1 + \frac{0.1794}{s^{0.95}} + 6.0722s^{0.17} \right] \left[ \frac{1}{1.23s^{0.05}} \right] \quad (16)$$

The parameters of the controller designed by Li et al. [27] are reported as:

$$C(s) = 1.3451 \left[ 1 + \frac{0.1283}{s^{0.95}} + 6.4475s^{0.17} \right] \left[ \frac{1 + 0.615s}{s^{0.05}(1.5051s^2 + 4.3716s + 5.3588)} \right] \quad (17)$$

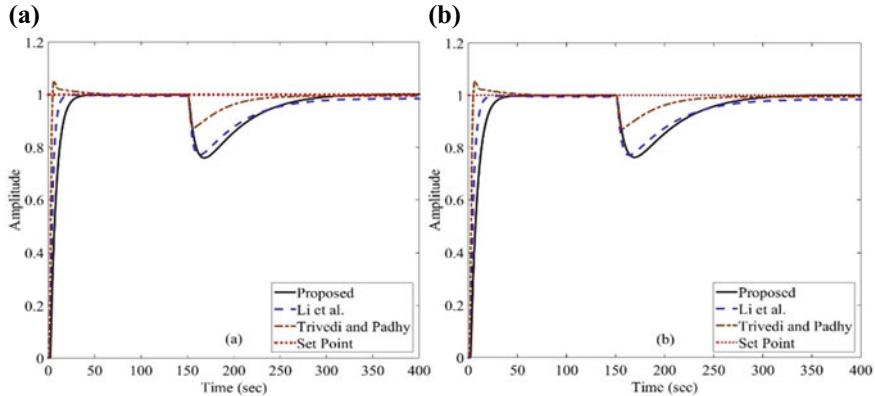
The nominal plant response is included in Fig. 4a and control actions corresponding to Fig. 4a are shown in Fig. 4b. Simulations are carried out by taking  $R(s) = 1$  at  $t = 0$  second and an external disturbance  $D(s) = -0.3$  at  $t = 150$  second. Table 3 displays a performance comparison for this study based on various errors. To demonstrate the viability of the proposed control law, simulations with 10% change in plant gain and 10% change in plant delay are conducted, and the corresponding responses are given in Fig. 5a, b.



**Fig.4** (a) Nominal response of plant of  $G_2(s)$  (b) Control inputs

**Table 3** ISE, ITAE, and IAE comparison of control performance for case study 2

Method	ISE	ITAE	IAE
Presented	6.086	2997	23.14
Li et al. [27]	5.557	3492	22.82
Trivedi and Padhy [28]	2.586	1210	9.395



**Fig. 5** Perturbed response of  $G_2(s)$  **(a)** 10% changes in gain **(b)** 10% changes in delay

### Case study 2: observation

Considering the results, the suggested strategy gives satisfactory servo tracking and regulatory action. The design of the suggested controller is very simple yet, a comparable performance is identified for the presented design with respect to the NIO controller by Trivedi and Padhy [26] and Li et al. [27]. The provided controller is able to deal with process uncertainties, as seen in Fig. 5a, b.

## 4 Conclusions

In this work, a modified FOTID controller based on direct synthesis approach is proposed to get improved reference tracking and perfect regulatory response. However, the arbitrary order controller design technique for non-integer order processes suffers from a lack of straightforward design methodology. In this study, a very simple and straightforward design methodology of a FOTID controller using direct synthesis is proposed. By simulating the DC servo system and bioreactor process models, the design's viability is confirmed, and it is observed that the proposed control rule consistently produces positive results. Additionally, the simulation results for perturbed processes guarantee their robust effectiveness when plant components vary. The suggested method accomplishes well in terms of errors, as shown in Tables 2 and 3.

## References

1. Chakraborty S, Naskar AK, Ghosh S (2020) Inverse plant model and frequency loop shaping-based PID controller design for processes with time-delay. *Int J Autom Control* 14(4):399–422
2. Chakraborty S, Singh J, Naskar AK, Ghosh S (2022) A new analytical approach for set-point weighted 2DOF-PID controller design for integrating plus time-delay processes: an experimental study. *IETE J Res*, 1–15
3. Das D, Chakraborty S, Lloyds Raja G (2022) Enhanced dual-DOF PI-PD control of integrating-type chemical processes. *Int J Chem Reactor Eng*
4. Sengupta S, Karan S, Dey C, Modified smith predictor-based optimal fractional PD-PD controller for double integrating processes with considerable time delay. *Asia-Pacific J Chem Eng*, e2873
5. Debnath L (2003) Recent applications of fractional calculus to science and engineering. *Int J Math Math Sci* 2003(54):3413–3442
6. Sun HongGuang, Zhang Y, Baleanu D, Chen W, Chen YangQuan (2018) A new collection of real-world applications of fractional calculus in science and engineering. *Commun Nonlinear Sci Numer Simul* 64:213–231
7. Kumar D, Aryan P, Lloyds Raja G (2022) Design of a novel fractional-order internal model controller-based Smith predictor for integrating processes with large dead-time. *Asia-Pacific J Chem Eng* 17(1):e2724
8. Zheng W, Chen Y, Wang X, Lin M, Guo J (2022) Robust fractional order PID controller synthesis for the first order plus integral system. *Measur Control*
9. Bettayeb M, Mansouri R, Al-Saggaf U, Mustafa Mehedi I (2017) Smith predictor based fractional-order-filter PID controllers design for long time delay systems. *Asian J Control* 19(2):587–598
10. Chakraborty S, Ghosh S, Naskar AK (2017) I-PD controller for integrating plus time-delay processes. *IET Control Theory Appl* 11(17):3137–3145
11. Chakraborty S, Ghosh S, Naskar AK (2017) All-PD control of pure integrating plus time-delay processes with gain and phase-margin specifications. *ISA Trans* 68:203–211
12. Das D, Chakraborty S, Naskar AK (2023) Controller design on a new 2DOF PID structure for different processes having integrating nature for both the step and ramp type of signals. *Int J Syst Sci*, 1–28
13. Meena R, Das S, Pal VC, Chakraborty S (2023) Smith-predictor based enhanced Dual-DOF fractional order control for integrating type CSTRs. *Int J Chem Reactor Eng*
14. Doostinia M, Beheshti MTH, Babaei M, Alavi SA, Ramezani A (2021) Control of main steam pressure in coal-fired power plant boilers by fractional-order controller with Smith predictor structure for delay compensation. *IETE J Res*, 1–15
15. Viola J, Angel L, Sebastian JM (2017) Design and robust performance evaluation of a fractional order PID controller applied to a DC motor. *IEEE/CAA J Automatica Sinica* 4(2):304–314
16. Saxena S, Hote YV (2022) Design of robust fractional-order controller using the Bode ideal transfer function approach in IMC paradigm. *Nonlinear Dyn* 107(1):983–1001
17. Arya PP, Chakrabarty S (2022) Robust modified partial internal model control for stable, unstable and integrating processes. *Int J Syst Sci*, 1–20
18. Mohan V, Chhabra H, Rani A, Singh V (2019) An expert 2DOF fractional order fuzzy PID controller for nonlinear systems. *Neural Comput Appl* 31(8):4253–4270
19. Shalaby R, El-Hossainy M, Abo-Zalam B, Mahmoud TA (2022) Optimal fractional-order PID controller based on fractional-order actor-critic algorithm. *Neural Comput Appl*, 1–34
20. Jakovljević B, Lino P, Maione G (2021) Control of double-loop permanent magnet synchronous motor drives by optimized fractional and distributed-order PID controllers. *Eur J Control* 58:232–244
21. Bingi K, Ibrahim R, Karsiti MN, Hassan SM, Harindran VR (2018) A comparative study of 2DOF PID and 2DOF fractional order PID controllers on a class of unstable systems. *Archives Control Sci*, 635–682

22. Pachauri N, Singh V, Rani A (2018) Two degrees-of-freedom fractional-order proportional–integral–derivative-based temperature control of fermentation process. *J Dyn Syst Meas Contr* 140(7):071006
23. Kumari S, Aryan P, Kumar D, Lloyds Raja G (2022) Hybrid dual-loop control method for dead-time second-order unstable inverse response plants with a case study on CSTR. *Int J Chem React Eng* 1:11
24. Prakash AP, Chakrabarty S (2020) A robust internal model-based fractional order controller for fractional order plus time delay processes. *IEEE Control Syst Lett* 4(4):862–867
25. Malek H, Luo Y, Chen YQ (2013) Identification and tuning fractional order proportional integral controllers for time delayed systems with a fractional pole. *Mechatronics* 23(7):746–754
26. Rishika T, Padhy PK (2020) Design of indirect fractional order IMC controller for fractional order processes. *IEEE Trans Circuits Syst II: Express Briefs* 68(3):968–972
27. Li D, Liu L, Jin Q, Hirasawa K (2015) Maximum sensitivity based fractional IMC–PID controller design for non-integer order system with time delay. *J Process Control* 31:17–29

# A Novel Modified Semicircular Patch Antenna for Next Generation Communication



Kiran Githala  and Manisha Gupta 

**Abstract** A neoteric semicircular antenna for improved wireless communication applications has been presented in this work. The semicircular patch is printed on a  $30 \times 30 \text{ mm}^2$  FR-4 dielectric substrate of thickness 1.59 mm. Bandwidth ratio of the proposed antenna is 14.38:1 for a VSWR < 2 and operates between the frequency range of 2.78 and 40 GHz. Proposed semicircular patch antenna's maximum gain is 5.61 dB. This antenna is suitable for Wi-Fi, GPS, Internet of Things applications, and other wireless technologies.

**Keywords** Semicircular shape · Super wideband · Bandwidth · Defected ground plane

## 1 Introduction

Technology in the modern communication system has advanced rapidly. Antenna is an essential element for a communication system, it acts as an input and output link for communication systems. It is a passive device that transforms a radio frequency into an electromagnetic wave in empty space. Antenna definitions according to the IEEE standard “a device for radiating and receiving radio waves” [1]. For next generation applications, the demand of an antenna of reduced size with good radiation efficiency and high gain is increased. According to Federal Communication Commission (FCC) regulation, “ultra-wideband antenna functions over a frequency 3.1 to 10.6 GHz with bandwidth ratio 3.4:1.” [2]. The bandwidth ratio for a super wideband is greater than or equal to 10:1 at  $-10 \text{ dB}$  return loss [3]. In last few years, a large number of super wideband antennas have been reported. The bandwidth of printed antennas has been improved by using many methods like fractal concept, various radiator shapes or slots, and several feeding mechanisms. As an illustration, K. R. Chen et al. presented a monopole antenna with dimensions of  $(35 \times 77) \text{ mm}^2$  that operates for the bandwidth range of 1.44 GHz to 18.8 GHz [4]. A new modified fractal monopole

---

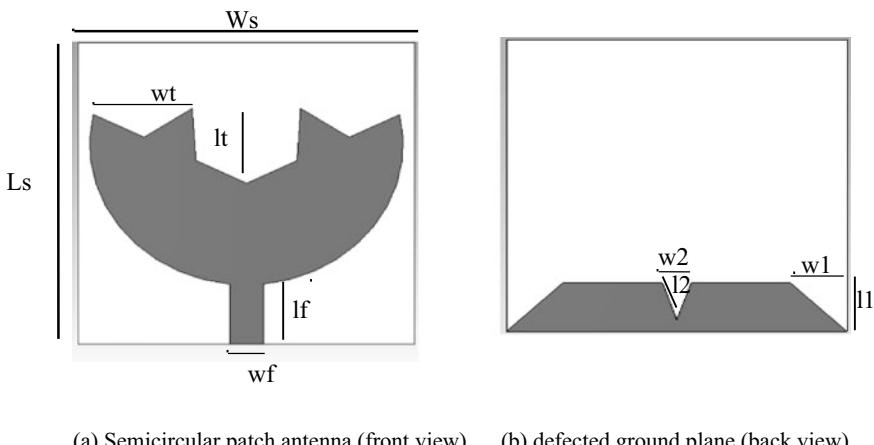
K. Githala · M. Gupta ()

Department of Physics, University of Rajasthan, Jaipur 302004, India

e-mail: [drguptamanisha@uniraj.ac.in](mailto:drguptamanisha@uniraj.ac.in)

antenna in star triangle shape was proposed by V. Waladi et al. for super wideband applications. It has a small dimension of  $(20 \times 20 \times 1)$  mm<sup>3</sup> and operates over the bandwidth of 1 GHz to 30 GHz [5]. A hexagonal seirpinski fractal antenna with dimensions of  $(30 \times 28)$  mm<sup>2</sup> that is CPW fed has been proposed by S. Singhal and A. K. Singh. This proposed antenna operates between 3.4 to 37.4 GHz [6]. A new modified U-shaped microstrip super wideband antenna with dimensions of  $(30 \times 30)$  mm<sup>2</sup> has been presented by M. Elhabachi et al. and this proposed antenna operates between 3 to 20 GHz, providing a nice radiation pattern and high gain [7]. For use in wireless sensor networks, R. Azim proposed a super wideband antenna that is CPW fed and has a bow tie shaped patch with the dimensions  $(25 \times 20)$  mm<sup>2</sup> [8]. A Mickey Mouse-shaped super wideband antenna with dimensions of  $(18 \times 18)$  mm<sup>2</sup> was presented by W. Balani et al. It operates for frequencies 1.22 to 47.5 GHz [9]. A  $(20 \times 20)$  mm<sup>2</sup> super wideband fractal antenna in star shape was presented by R. Malik et al. for use in 5G applications. It is functional from 17.22 to 180 GHz [10]. A  $(50 \times 50)$  mm<sup>2</sup> fractal ring-shaped ultra-wideband antenna was proposed by M. Rahimi et al. It is beneficial for notch applications and operates over the range of 2.1 to 11 GHz [11].

This article presents a compact semicircular patch super wideband antenna. The compact semicircle shaped antenna measures  $30 \times 30$  mm<sup>2</sup> in size and is printed on FR-4, which has a 1.59 mm thickness. The designed semicircular patch antenna operates between 2.78 and 40 GHz. The proposed antenna's frequency range satisfies the need for increased wireless communication by providing a wider bandwidth.

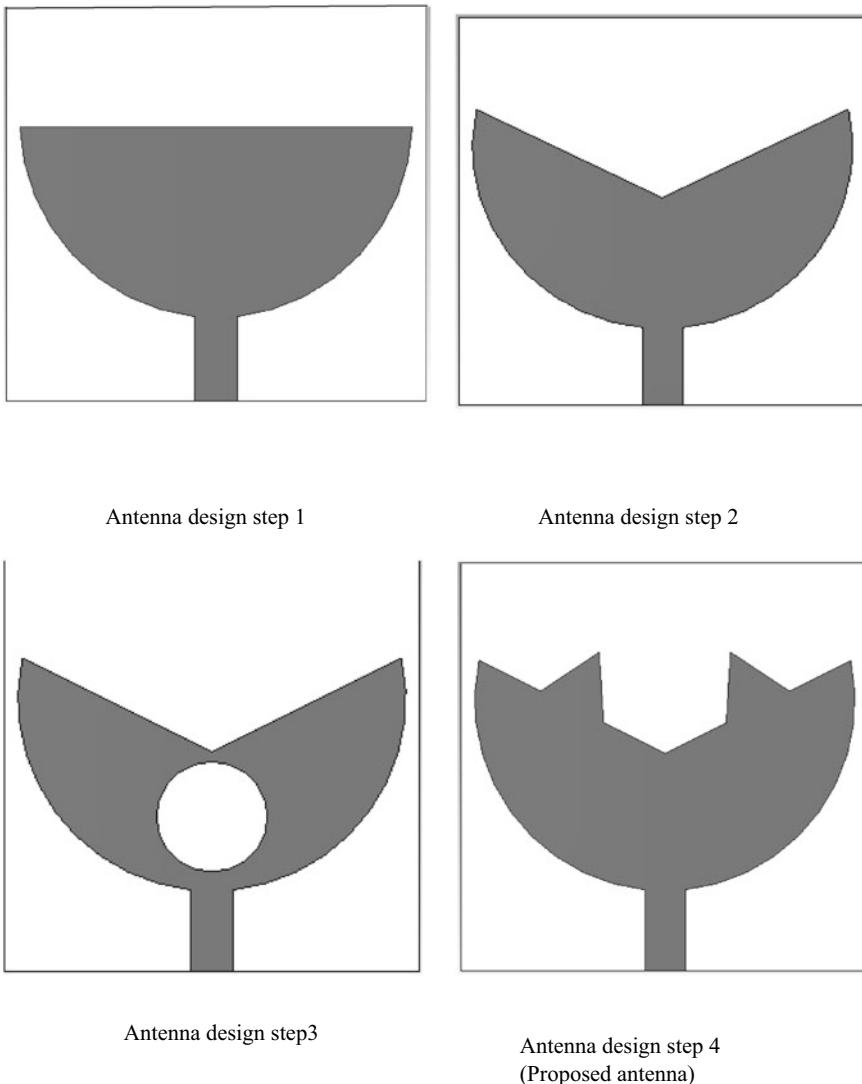


(a) Semicircular patch antenna (front view) (b) defected ground plane (back view)

**Fig. 1** Semicircular patch antenna's design and parameters [ $W_s = L_s = 30$ ,  $w_f = 3$ ,  $l_f = 5.9$ ,  $w_t = 14$ ,  $l_t = 7$ ,  $w_1 = 5$ ,  $l_1 = 5$ ,  $w_2 = 2.5$ ,  $l_2 = 4$  (units in mm)]

## 2 Antenna Design

Figure 1 depicts the proposed antenna's parameters and design. This antenna uses a microstrip feed line and is printed on a  $30 \times 30 \text{ mm}^2$  FR-4 substrate. The FR-4 substrate is 1.59 mm thick, having a dielectric const. of 4.3, and a loss tangent of 0.025. The proposed antenna's compact shape ( $30 \times 30 \text{ mm}^2$ ) is based on a U-shaped printed monopole antenna using the example shown in [7, 12].



**Fig. 2** Semicircular patch antenna's design steps

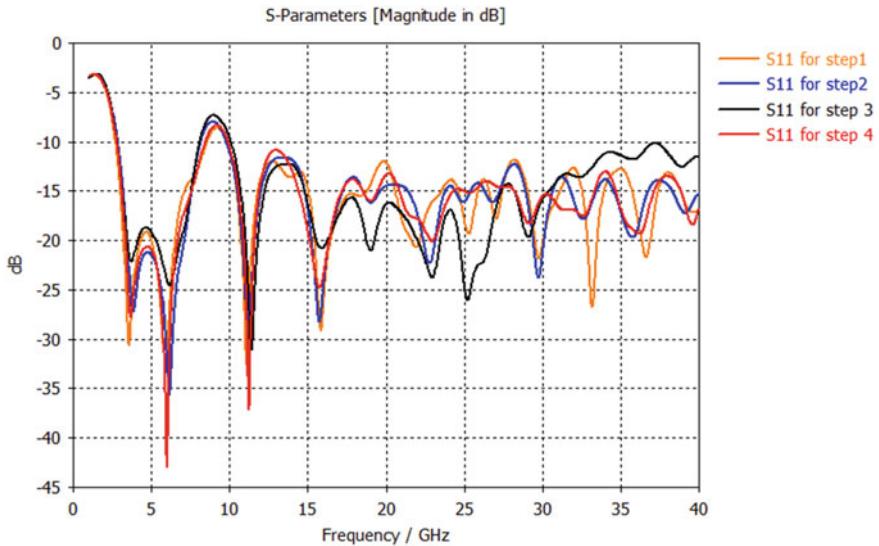
**Table 1** Antenna parameter values

Parameter	Description	Size (mm)
Ws	Antenna's width	30
Ls	Antenna's length	30
r	Radius of patch	14
wt	Triangular slot's width	14
lt	Triangular slot's length	7
lg	Ground plane length	30
lf	Feed line length	5.9
wf	Feed line width	3
w1	Triangular slot's width on ground plane	5
l1	Triangular slot's length on ground plane	5

The proposed antenna's semicircular patch structure is printed on FR-4 substrate. To increase the antenna's bandwidth, both ends of semicircular patch have been designed into a triangular groove. Figure 2 depicts the proposed antenna's design process. To achieve wide bandwidth range the proposed semicircular patch antenna's ground plane has tapered on the edges. Hence, the reflection coefficient achieved for wide frequency range becomes less than  $-10$  dB. Figure 2 depicts the design steps of semicircular patch antenna (Table 1).

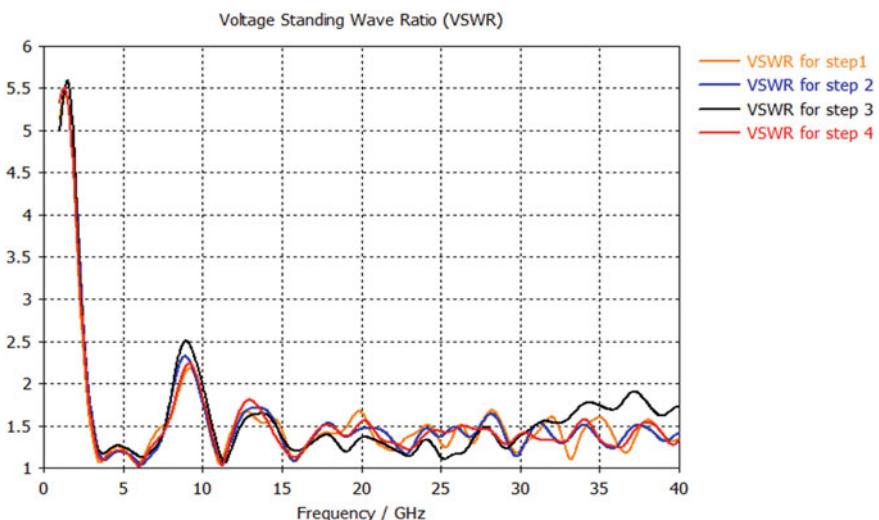
### 3 Results and Discussions

Simulation process of semicircular patch antenna is done by using CST microwave studio software. Figure 3 displays the comparison of return losses ( $S_{11}$ ) between different semicircular patch antennas. By comparing these results, frequency shift is observed in every step. Designed antenna in step 1 operates for 2.6 to 40 GHz and has a minimum return loss of  $-32.44$  dB at 11.06 GHz. Semicircular patch antenna in step 2 operates for a bandwidth range of 2.81 to 40 GHz and has minimum return loss of  $-35$  dB at 6.14 GHz. The antenna in Step 3 has a bandwidth of 2.79 to 37.201 GHz with minimum return loss of  $-30$  dB at 11.359 GHz. Semicircular patch antenna in step 4 (proposed antenna) has bandwidth range of 2.78 to 40 GHz and minimum return loss of  $-42.901$  dB at 5.99 GHz, making it the design with the least loss, and thus, best results.



**Fig. 3** Comparison between different  $S_{11}$  for designed antenna

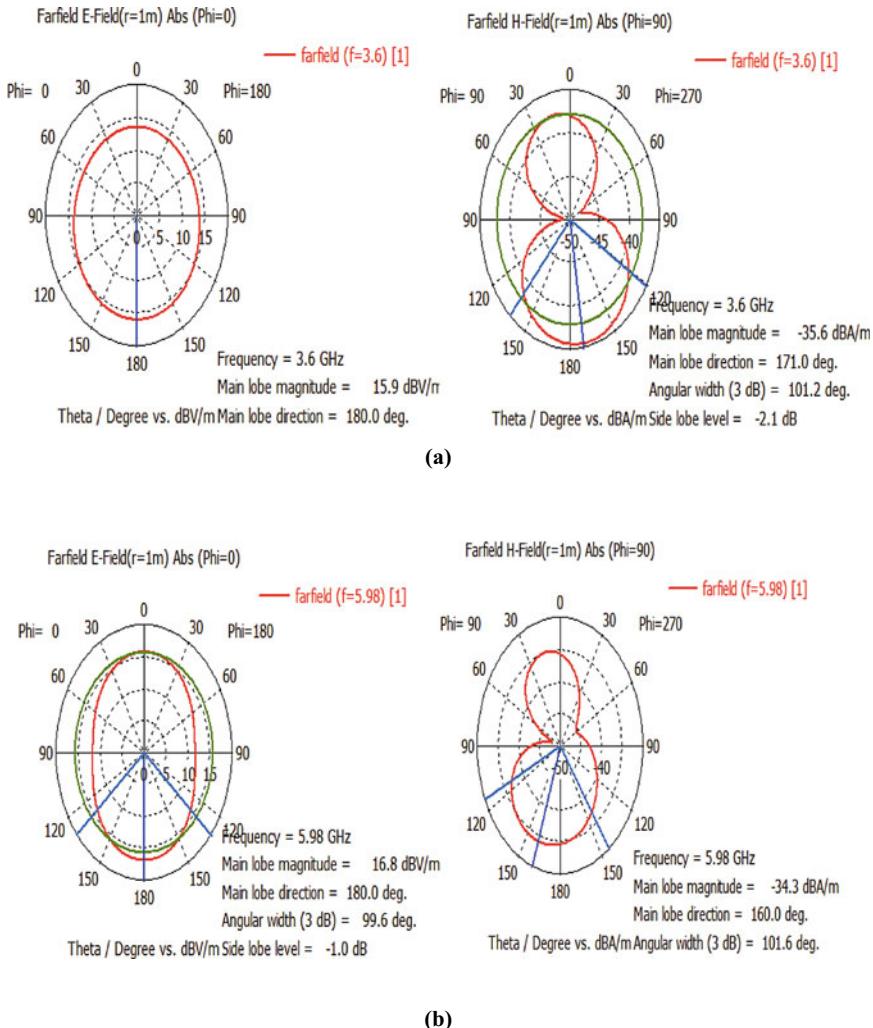
Figure 4 displays comparison of Voltage Standing Wave Ratio (VSWR) between the designed semicircular patch antennas. VSWR of the designed semicircular patch antenna in step 4 shows the best results among these designed antennas, with its value less than 2 for frequency ranges 2.73 to 8.59 GHz and 9.71 to 40 GHz.



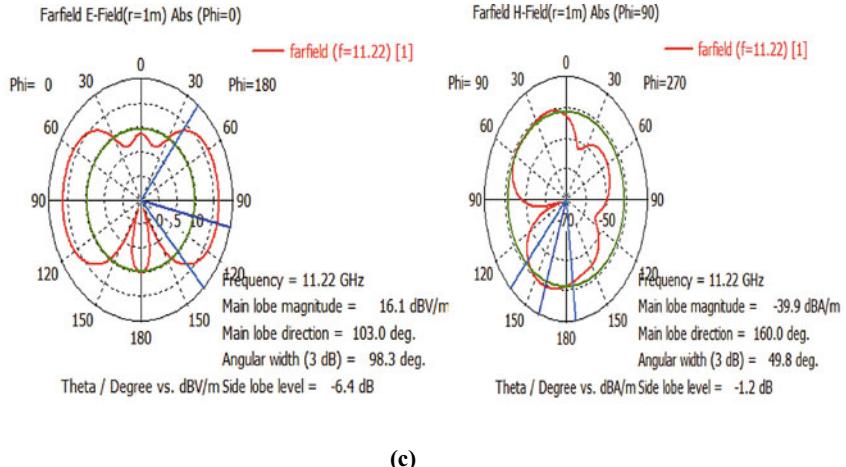
**Fig. 4** Comparison between simulation results of semicircular patch antenna “VSWR versus frequency graph”

Figure 5 depicts the proposed semicircular patch antenna's simulated radiation patterns for  $\phi = 0^\circ$  and  $\phi = 90^\circ$  at 3.6 GHz, 5.98 GHz, 11.22 GHz, and 15.78 GHz.

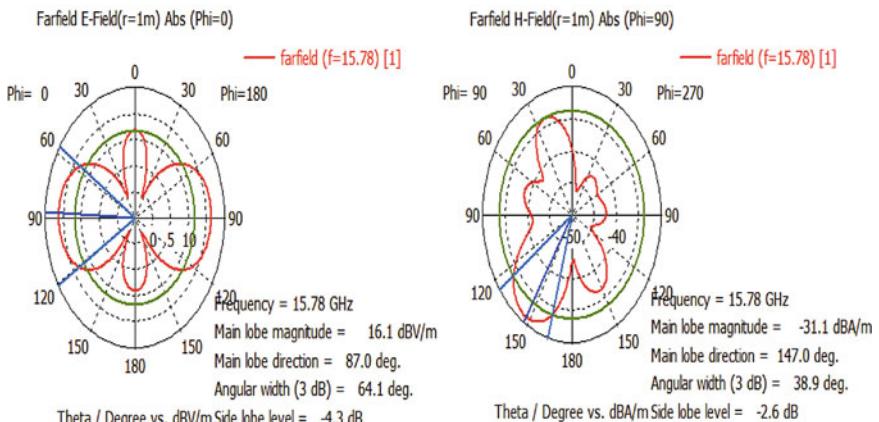
Figure 6 depicts the gain graph for the designed semicircular antenna. Here, the gain of the proposed semicircular patch antenna increases from 1.81 dB to 5.61 dB. The proposed antenna has maximum gain of 5.61 dB at 15.78 GHz.



**Fig. 5** Radiation patterns for E and H plane for proposed semicircular antenna at frequencies **a** 3.6 GHz, **b** 5.98 GHz, **c** 11.22 GHz, and **d** 15.78 GHz



(c)

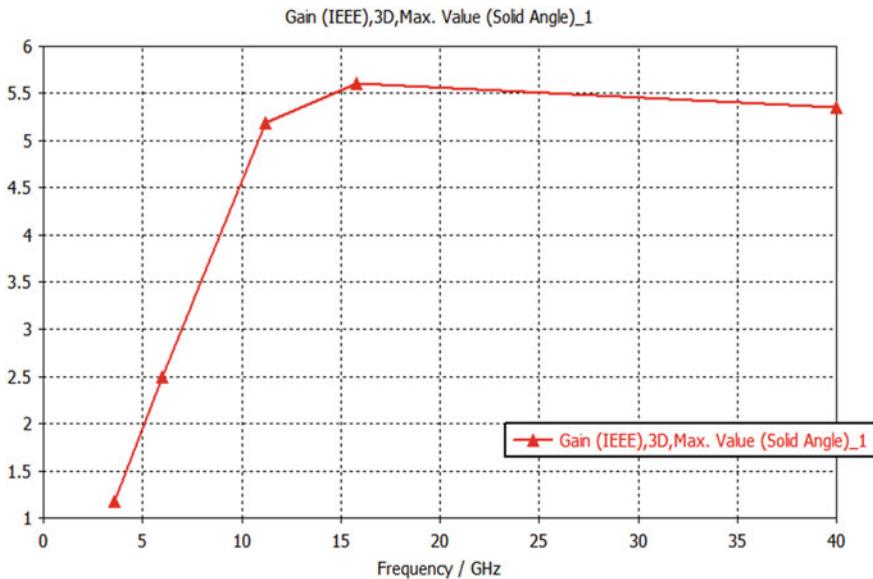


(d)

 $\phi = 0^\circ$  $\phi = 90^\circ$ **Fig. 5** (continued)

#### 4 Comparison Between Previously Designed Antennas and Proposed Antenna

The proposed antenna, due to its compact size, large bandwidth, and  $VSWR < 2$  gives good results for super wideband applications. Table 2 gives a comparison between previously designed super wideband antennas and currently proposed antenna based on various parameters.



**Fig. 6** Proposed semicircular antenna's simulated gain versus frequency graph

**Table 2** Comparison between existing super wideband and ultra-wideband antennas with proposed antenna

References	Size (mm <sup>2</sup> )	Bandwidth (GHz)	Band width ratio
Proposed	30 × 30	2.78–40	14.38:1
[13]	52 × 42	0.96–13.98	14.56:1
[14]	32 × 22	2.5–29	11.6:1
[15]	27 × 32	2.56–39.47	15.41:1
[16]	135 × 135	1–19.4	19.4:1
[17]	25 × 30	2.43–70	28.8:1
[18]	50 × 50	2.5–12	4.8:1
[19]	30 × 35	2.25–12.8	5.68:1
[20]	49 × 54.5	2.6–22	8.46:1

## 5 Conclusion

A semicircular patch ultra-wideband antenna has been designed in this work. This designed antenna covers a wideband from 2.78 to 40 GHz frequencies. Simulated results in comparison with other existing and proposed antennas show empirically that the designed semicircular patch antenna is a suitable choice to achieve the needs of next generation communication applications. An antenna's bandwidth can be increased using the defective ground concept. 5.61 dB Maximum gain is seen by the proposed antenna. The designed semicircular patch antenna is suitable for Wi-Fi, GPS, Internet of Things applications, and other wireless technologies.

## References

1. IEEE standard definition of terms of antenna, IEEE STD 145 (1983)
2. Federal Communication Commission. Revision of part 15 of the commission's rules regarding ultra-wide band transmission systems first report and order FCC 02, V48, Washington (2002)
3. Balani W, Sarvagya M, Ali T, Pai MMM, Anguera J, Andujar A, Das S (2019) Design techniques of super-wideband antenna- existing and future prospective. *IEEE Access* 7:141241–141257
4. Chen KR, Sim CYD, Row JS (2011) A compact monopole antenna for super wideband applications. *IEEE Antennas Wirel Propagation Lett* 10:488–491
5. Waladi V, Mohammadi N, Zehforoosh Y, Habashi A, Nourinia J (2013) A novel modified star-triangular fractal (MSTF) monopole antenna for super-wideband applications. *IEEE Antennas Wirel Propagation Lett* 12:651–654
6. Singhal S, Singh AK (2016) CPW-fed hexagonal Sierpinski super wideband fractal antenna. *IET Microwave Antennas Propagation* 10:1701–1707
7. Elhabchi M, Srfi MN, Touahni R (2020) A novel modified U-shaped micro strip antenna for super wide band applications. *Analog Integr Circ Sig Process* 102:571–578
8. Azim R, Islam MT, Arshad H, Alam MM, Sobahi N, Khan AI (2021) CPW- fed super wideband antenna with modified vertical bow-tie-shaped patch for wireless sensor networks. *IEEE Access* 9:5343–5353
9. Balani W, Sarvagya M, Samasgikar A, Ali T, Kumar P (2021) Design and analysis of super wideband antenna for microwave applications. *Sensors* 21:477
10. Malik R, Singh P, Ali H, Goel T (2018) A star shaped super wide band fractal antenna for 5G applications. In: 3rd International conference for convergence in technology (I2CT), pp 1–6, Pune, India
11. Rahimi M, Heydari S, Mansouri Z, Gandji NP, Zarrabi FB (2016) Investigation and design of an ultra-wideband fractal ring antenna for notch applications. *Microw Opt Technol Lett* 58:1629–1634
12. Jangid KG, Jain PK, Sharma BR, Saxena VK, Kulhar VS, Bhatnagar D (2017) Ring slotted circularly polarized U-shaped printed monopole antenna for various wireless applications. *Adv Electromagnetics* 6:70–76
13. Rahman MN, Islam MT, Mahmud MZ, Samsuzzaman M (2017) Compact micro strip patch antenna proclaiming super wideband characteristics. *Microwave Opt Technol Lett* 59:2563–2570
14. Okas P, Sharma A, Das G, Gangwar RK (2018) Elliptical slot loaded partially segmented circular monopole antenna for super wideband application. *AEU-Int J Electron Commun* 88:63–69
15. Sharma M, Awasthi YK, Kaul C (2019) Design and study of super wideband monopole antenna for various wireless applications including dual notched band characteristics. In: 6th International conference on signal processing and integrated networks (SPIN), pp 82–86, IEEE
16. Yeo J, Lee JI (2014) Coupled-sectorial-loop antenna with circular sectors for super wideband applications. *Microwave Opt Technol Lett* 56:1683–1689
17. Sediq HT, Nourinia J, Ghobadi C (2022) A cat-shaped patch antenna for future super wideband wireless microwave applications. *Wirel Pers Commun* 125:1307–1333
18. Vendik IB, Rusakov A, Kanjanasit K, Hong J, Filonov D (2017) Ultra-wideband (UWB) planar antenna with single-, dual-, and triple-band notched characteristic based on electric ring resonator. *IEEE Antennas Wirel Propagation Lett* 16:1597–1600
19. Ebadzadeh SR, Zehforoosh Y (2017) A compact UWB monopole antenna with rejected WLAN band using split-ring resonator and assessed by analytic hierarchy process method. *J Microwave Optoelectron Electromagnetic Appl* 16:592–601
20. Kumar A, Saxena G, Kumar P, Awasthi YK, Jain P, Singhwal SS, Ranjan P (2022) Quad-band circularly polarized super-wideband MIMO antenna for wireless applications. *Int J RF Microwave Comput Aided Eng* 32:1–21

# Panoptic Driving Perception Using Deep Learning Techniques



**Uday Kulkarni, Shashank Agasimani, Sagar Kabadi, P. S. Aditya, Vittal Hundarad, and Rajeshkanna Viswanathan**

**Abstract** The technology of autonomous driving allows vehicles to operate independently without the intervention of a human driver. This system relies on sensors and algorithms to execute driving functions, such as braking, steering and accelerating, by interpreting the vehicle's environment and making decisions accordingly. A critical component of the autonomous driving system is the panoptic driving perception system, which offers a comprehensive understanding of the surrounding environment to the vehicle, leading to advancements in autonomous driving technology. We introduce a panoptic driving perception network that performs drivable area segmentation, traffic object detection, lane detection and distance computation concurrently. The model includes three decoders for specialized tasks and an encoder for feature extraction. The model has delivered exceptional results on the demanding BDD100K dataset by achieving state-of-the-art performance in speed and accuracy for all four tasks, making it the first study to process these four simultaneous visual perception tasks while maintaining high precision. Traditionally, Ultrasonic sensors like HC-sr04 have been used to measure the distance of objects. However, this approach may not be practical when designing embedded devices requiring both object detection (via a camera) and distance measurement capabilities. Fortunately, there is an alternative approach to calculate the distance of an object from the camera by using the depth information utilized for object detection through bounding boxes.

**Keywords** Autonomous driving system · Panoptic driving perception · You only look once · Area segmentation · Lane detection · Distance computation

---

U. Kulkarni (✉) · S. Agasimani · S. Kabadi · P. S. Aditya · V. Hundarad · R. Viswanathan  
School of Computer Science and Engineering, KLE Technological University, Hubballi 580031,  
Karnataka, India  
e-mail: [uday\\_kulkarni@kletech.ac.in](mailto:uday_kulkarni@kletech.ac.in)

## 1 Introduction

Autonomous driving [9] refers to the ability of a vehicle to operate and navigate without the need for human input or intervention. It relies on various sensors, technologies and algorithms to sense the surrounding environment, detect obstacles, interpret road signs and make decisions on acceleration, braking and steering. Autonomous driving systems [20] have the potential to improve road safety, reduce traffic congestion and provide greater mobility for people who are unable to drive themselves. Despite the significant advancements in deep learning [12] and computer vision [17], vision-based tasks [13] continue to pose challenges in low-cost autonomous driving applications [18]. Therefore, the importance of panoptic driving perception [23] cannot be overstated in the development of effective and reliable autonomous driving systems [18].

A panoptic driving perception system [9] is a complex system of sensors, cameras, light detection and ranging, radars and other technologies that work together to provide a comprehensive view of the vehicle's surroundings. The system captures and processes data from different sources to create a high-resolution 3D map [8] of the environment in real-time. This map includes information about other vehicles, pedestrians, road markings, signs, traffic lights and obstacles that may be present on the road. The system then uses this information to provide the vehicle with a clear understanding of its surroundings, allowing it to make informed decisions on driving manoeuvres, such as steering, acceleration and braking. The accuracy and reliability of the panoptic driving perception [17] system are crucial for the safe and effective operation of autonomous vehicles [9].

Object detection is a computer vision [17] task in machine learning [7] that involves identifying and localizing objects within an image or video. The objective is to classify the objects in the image and locate them with a bounding box [5]. Object detection involves two primary tasks, Classification [22] and Localization. Classification refers to the process of identifying the category of an object in an image. Localization [10] involves predicting the coordinates of a bounding box that surrounds the object of interest. The accurate detection of objects is crucial in providing position and size information of traffic obstacles, which is essential for autonomous vehicles [17] to make timely and precise decisions while driving. Additionally, the segmentation of drivable areas and lanes provides valuable details for strengthening driving safety and route planning. Hence, lane segmentation [1], drivable area segmentation and object detection collectively play a critical role in enhancing the effectiveness and safety of autonomous driving systems [23].

Various techniques approach these tasks as distinct processes. For example, object detection [1] is handled by YOLO [5] and Faster R-CNN [18], Semantic Segmentation [19] is performed by PSPNet and ENet, and Lane detection is carried out by SADENet and SCNN. Although these methods demonstrate impressive performance, performing these tasks sequentially takes more time than addressing them simultaneously. Therefore, while implementing the panoptic driving perception [17] on embedded devices [18] that are typically used in self-driving cars, it is important

to consider restricted computational resources and latency. Moreover, many of the tasks involved in comprehending traffic scenes have interconnected information that should also be taken into consideration. Thus, after conducting lot of comprehensive analysis of prior approaches, we have introduced a productive and streamlined multi-task learning [12] network in this paper.

We conducted experiments on the challenging BDD100K dataset [23] to address the Panoptic driving perception multitasking issue [9], which involves detecting traffic objects, segmenting [19] drivable areas and lanes, and computing distances. Our approach uses a lightweight CNN as an encoder to draw out image features, followed by three decoders for each respective task. We employed a single-stage detection network for object detection [21], which is faster and more closely in connection with the other two semantic segmentation [4] tasks. Our segmentation branch leverages semantic attributes [17] of varying scales and levels from the encoder's feature map to achieve high pixel-wise semantic prediction [3] accuracy while maintaining fast processing speed. Our model has achieved excellent accuracy in all the four tasks.

Our Primary contributions include designing a strong network architecture to train the model, it has been generalized to adapt to diverse scenarios while also maintaining high processing speed, Quicker (network structure and memory allocation [9] strategy that is more efficient) and Better (more efficient model structure and incorporated advanced techniques, such as Mosaic and Mix-up, into the data pre-processing stage and novel hybrid loss to improve the overall performance of the model) attaining 84.6% Mean Average Precision (MAP) [24] for object detection task, 88% Mean Intersection of Union (MIoU) [7] for the drivable area segmentation [17] task, and 75.8% accuracy for lane detection and 88.09% for distance computation.

## 2 Literature Survey

“Research on the Cascade Vehicle Detection Method Based on CNN” [23] a paper on research and development in the field of computer vision for automated driving systems. One of the significant challenges in developing such systems is the accurate detection of lane markings on the road. Traditional lane detection methods have relied on feature extraction and high-definition imaging, which can be computationally intensive and may not be able to handle real-world scenarios. However, by using cloud computing and edge computing, it is possible to improve the efficiency of lane detection and overcome some of these limitations. Cloud computing can be used to offload the processing of lane detection tasks to remote servers, which have access to vast amounts of computing resources. This can help reduce the computational load on the local system and enable more efficient processing of the data.

Edge computing, on the other hand, involves performing computing tasks closer to the source of the data. In the context of lane detection, this means processing the image data at the edge of the network, closer to the camera or sensor that captures it. This can help reduce latency and improve the responsiveness of the system. A dual model based on instance segmentation and using convolutional neural networks (CNNs) can

be used to enhance the accuracy of lane detection. Instance segmentation involves not only detecting the lane markings but also segmenting each individual lane marking instance in the image. This can help improve the accuracy of lane detection and enable more precise control of the vehicle. In summary, the combination of cloud computing and edge computing, along with advanced computer vision techniques like instance segmentation and CNNs, can help improve the accuracy and efficiency of lane detection in automated driving systems.

“YOLOP : You Only Look Once for Panoptic Driving Perception” [9] present a research in which panoptic driving perception system is considered a crucial component of autonomous driving. Such a system should be able to accurately and rapidly perceive the surrounding environment, including detecting traffic objects, identifying drivable areas and detecting lane markings. The YOLOP network is a novel approach to panoptic driving perception. It is a multi-task learning model that simultaneously performs drivable area segmentation, traffic object detection and lane detection. The YOLOP network consists of one encoder for feature extraction and three decoders, each of which is dedicated to handling one of the three specific tasks. The YOLOP network has demonstrated excellent performance on the challenging BDD100K dataset, achieving state-of-the-art results in terms of accuracy and speed on all three tasks. The effectiveness of the YOLOP network has been verified through ablative studies, which involve systematically removing components of the network to determine their impact on performance.

Overall, the YOLOP network represents a significant advancement in panoptic driving perception systems. Its ability to perform multiple tasks simultaneously with high accuracy and speed makes it a promising candidate for use in autonomous driving systems.

“Automatic vehicle detection system in Day and Night Mode: challenges, applications and panoramic review” [20] present the development of an efficient automatic onboard driver assistance system. Detecting vehicles at night is a challenging task due to various factors like low illumination, weather conditions like rain or snow, and other challenges. In recent years, many researchers have proposed different methodologies for detecting day and night vision vehicles, and this research paper provides a comprehensive review of the work done thus far. The paper discusses various vehicle detection methods and their applications in intelligent transport systems. It also includes a brief assessment of the documented techniques for identifying various types of vehicles in various surroundings, as well as the obstacles encountered by other researchers in their research area.

Some of the methods discussed in the paper include feature-based methods, deep learning-based techniques and image processing-based techniques. The feature-based methods involve extracting relevant features from the images and then using these features to classify the vehicles. The deep learning-based methods are used to extract characteristics from images using neural networks and then classify the vehicles. The image processing-based methods involve processing the images to enhance the features of the vehicles and then classify them. The paper also discusses the role of intelligent transport systems in the application of vehicle detection and recognition. Intelligent transport systems use various technologies such as sensors,

cameras and communication systems to provide real-time data about the traffic flow, road conditions and other relevant information. This information can then be used to improve the efficiency of traffic management systems.

In conclusion, vehicle detection and recognition is a challenging task, especially at night, and various methods have been proposed to address this issue. The paper provides a complete evaluation of the work done so far and the obstacles experienced by other scholars in their subject area. The importance of intelligent transportation systems in vehicle detection and recognition is also explored.

“Testing DNN-based Autonomous Driving Systems under Critical Environmental Conditions” [17] present a new approach, called TACTIC, in which DNN-based autonomous driving systems (ADS) are put into the test under different environmental conditions. According to the their research, evaluating such systems is becoming increasingly important due to the rising use of DNN-based ADS, where unexpected or erroneous behaviour might lead to catastrophic failures. Current systems frequently focus just on detecting incorrect behaviour and have not adequately investigated the influence of environmental factors.

### 3 Proposed Methodology

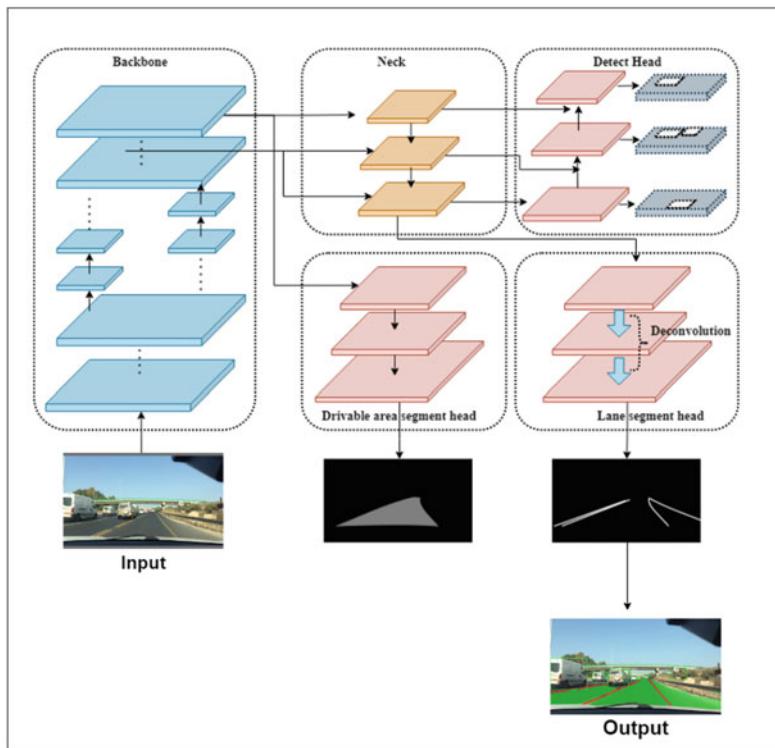
We propose a feed-forward network that is both uncomplicated and effective in achieving drivable area segmentation [9], traffic object detection [11] and lane detection [5] tasks simultaneously and distance is computed using the formula described below. The network features a shared encoder [2] and three dedicated decoders [2] for each task. The absence of intricate shared blocks between decoders [16] minimizes computational costs and simplifies end-to-end training of the network (Fig. 1).

#### 3.1 Encoder

Our designed network has a single encoder [7] that is split into a neck and a backbone network [16].

##### 3.1.1 Backbone

The features of the input image are extracted using the backbone network. Typically, the foundation is made up of a few traditional image categorization networks [3]. Because YOLOv8 [9] performs so well at detecting objects, we decide to use it because it reduces gradient duplication during optimization, CSPDarknet [7] is used as the backbone. The number of parameters and calculations is decreased since feature propagation [12] and reuse is supported. As a result, it helps to guarantee the network’s real-time performance.



**Fig. 1** Proposed architecture

### 3.1.2 Neck

The features of the input image are extracted using the backbone network. Typically, the foundation is made up of a few traditional image categorization networks [18]. Because YOLOv8 performs so well at detecting objects, we decide to use CSPDarknet [18] as the backbone because it eliminates the issue of gradient duplication during optimization. The number of parameters and calculations is decreased since it supports feature reuse [1] and feature propagation [2]. As a result, it helps to guarantee the network's real-time performance.

## 3.2 Decoders

Three specific tasks are being performed by three heads designed by our networks.

### 3.2.1 Detection of Head

Our detection approach uses an anchor-based multi-scale detection [11] method that resembles YOLOv8. We begin by utilizing a bottom-up feature pyramid [10] network called the Path Aggregation Network (PAN) [18], which transfers positional features from the bottom-up while FPN transfers semantic features top-down [19]. To enhance feature fusion, we combine them. For detection, we employ multi-scale fusion feature maps in PAN [21]. Using the multi-scale feature map [20], our detecting head predicts the position offset [24], height and width scaling, category probability, and prediction confidence [2] by assigning three previous anchors with different aspect ratios to each multi-scale feature map grid [5].

### 3.2.2 Drivable Area Segment and Lane Line Segment Head

The network structure of both the drivable area segment [14] head and the Lane line Segment head [2] is identical. The segmentation branch [7] takes the bottom FPN layer [6] as input, with dimensions of (W/8, H/8, 256). The segmentation process [3] is simple and involves three methods of upsampling [24], resulting in an output feature map with dimensions of (W, H, 2). The result reflects the likelihood that each pixel in the source/input image is a part of the drive-able region/laneline [6] or the background. Our network's performance is not enhanced by adding an additional SPP module to the segment branches [24], as is typically done by others, because of the shared SPP in the neck network. To reduce computation costs, we use the closest interpolation [3] method instead of deconvolution in our up-sample layer. Consequently, our segment decoders [1] produce highly precise output and perform inferences quickly.

Also, in order to save on computation costs, we substitute the Closest Interpolation [13] method for deconvolution [22] in our up-sample layer. As a result, our segment decoders not only produce output with great precision but also do inference very quickly.

## 3.3 Loss Function

Given that our network has three decoders [1], the multi-task loss function [6] comprises three components. Specifically, the detection loss ( $\mathcal{L}_{det}$ ) is determined by a combination of the object loss ( $L_{obj}$ ), classification loss ( $L_{class}$ ), and bounding box loss ( $L_{box}$ ), as depicted in the provided equation.

In order to minimize the loss of well-classified examples and focus the network on the most difficult ones, we utilize focal loss for  $L_{class}$  and  $L_{obj}$ .  $L_{class}$  is employed for correcting classification failures, while  $L_{obj}$  penalizes the confidence of a single prediction. For  $L_{box}$ , we use  $L_{IoU}$  [3], which considers the separation, overlap rate

[22], similarity of size, and aspect ratio between the predicted box and the ground truth [7].

Furthermore, in order to reduce classification failures between the targets and the network outputs [8] we do this by comparing the pixels of the network output and the targets, both the drivable area segmentation loss ( $L_{da-seg}$ ) [22] and lane line segmentation loss ( $L_{ll-seg}$ ) [22] including cross entropy loss with logits ( $L_{ce}$ ) [22].

To summarize, our ultimate loss function [6] is a combination of the three parts, weighted accordingly, The values of  $\alpha_1$ ,  $\alpha_2$ ,  $\alpha_3$ ,  $\gamma_1$ ,  $\gamma_2$  and  $\gamma_3$  can be adjusted to achieve a balance between all components of overall loss.

$$\mathcal{L}_{det} = \alpha_1 \mathcal{L}_{class} + \alpha_2 \mathcal{L}_{obj} + \alpha_3 \mathcal{L}_{box} \quad (1)$$

$$\mathcal{L}_{da-seg} = \mathcal{L}_{ce} \quad (2)$$

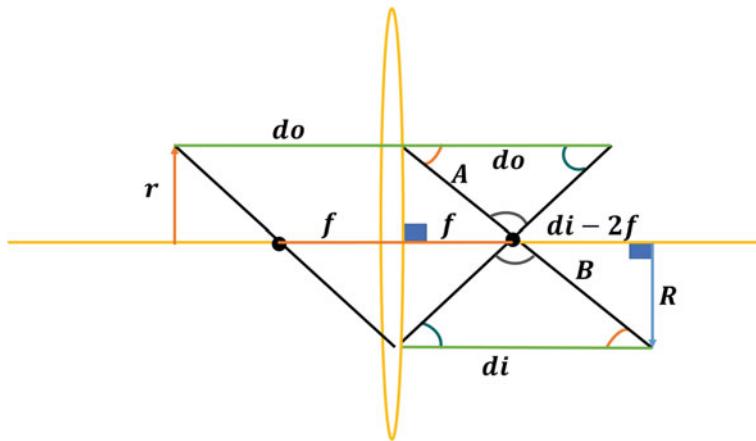
$$\mathcal{L}_{ll-seg} = \mathcal{L}_{ce} + \mathcal{L}_{IoU}. \quad (3)$$

$$\mathcal{L}_{all} = \gamma_1 \mathcal{L}_{det} + \gamma_2 \mathcal{L}_{da-seg} + \gamma_3 \mathcal{L}_{ll-seg} \quad (4)$$

### 3.4 Distance Computation

The objects detected in input images contain bounding box [17] for each object detected in it and it is represented by 4 numbers—(x0, y0, width, height). The x0 and y0 values are used to adjust or shift the bounding box [3]. The width and height values provide details about the detected object, such as its size and shape, and may vary depending on the object’s distance from the camera. Images passing through a lens are refracted, causing a slight stretch in the resulting image. This is in contrast to mirrors [7], which produce an exact reflection of the image. The amount of stretch depends on the angle at which the light enters the lens, as shown in following image

The green line in the diagram represents the actual distance between an object and a convex lens, while the “di” [14] line provides information about the appearance of the image. To understand this concept, imagine a triangle on the left side of the refracted image, with a base of “do”. By creating a similar triangle with the same perpendicular distance, we can observe that the two triangles on the right side are also similar, with “do” and “di” being parallel to each other. This means that the ratio of their corresponding sides is equivalent, which is expressed as  $do/di = A/B$ . Furthermore, when comparing the two triangles on the right side, with opposing angles that are equal and a right angle in each triangle. The hypotenuse of the comparable triangle, when both triangles have a right angle, is represented by the A:B ratio (as shown in the dark blue area of the Fig. 2). Thus, the new equation can be represented as



**Fig. 2** Relation between distance and focal length

$$\frac{do}{di} = \frac{A}{B} = \frac{f}{di - f} \quad (5)$$

The Simplified formula for computation of distance for detected bounding box is given below

$$\text{Distance} = (2 \times 3.14 \times 180) \div (w + h \times 360) \times 1000 + 3 \quad (6)$$

### 3.5 Experimental Setup

#### 3.5.1 Dataset Description

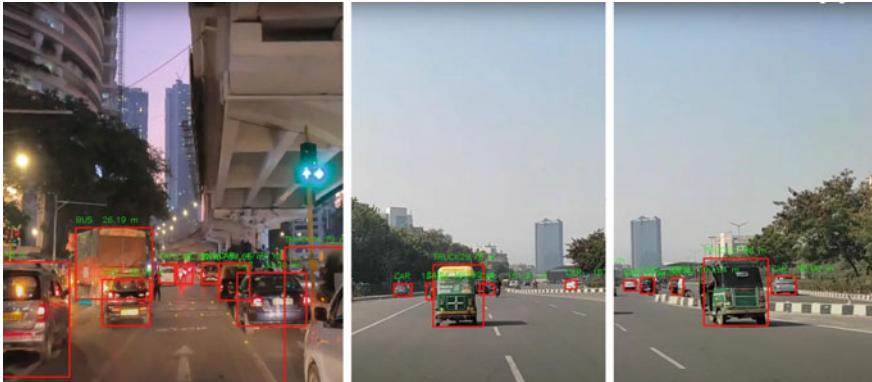
The BDD100K dataset [11] is an extensive resource for exploring multi-task learning in the autonomous driving domain. It comprises 100,000 images with annotations [5] for ten different tasks, making it the largest video dataset for driving. This dataset's diversity, encompassing various geographies, environments and weather conditions, ensures that any algorithm trained on it is resilient enough to adapt to a variety of environments. Hence, BDD100K dataset [23] has been selected to train and assess our network. The dataset comprises a training set with 70,000 images, a validation set with 10,000 images and a test set with 20,000 images. We analyse our network's performance on the validation set [15] instead because the labels on the test set are private.

### 3.5.2 Experiment Settings

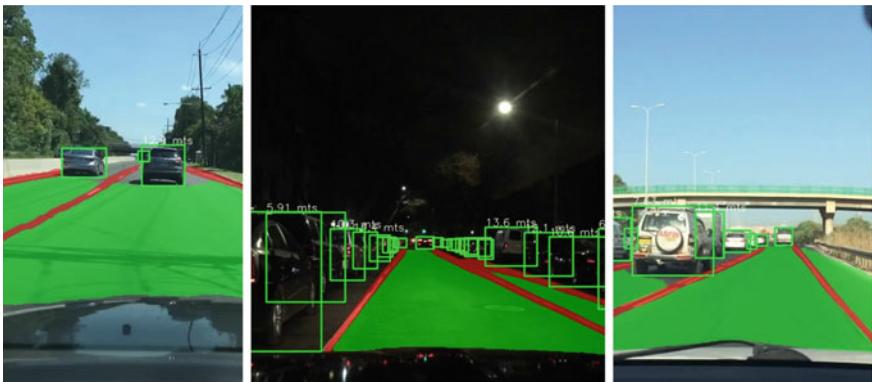
We have conducted a comparison of our network with several high-performing networks designed for both multitasking and individual tasks. Our focus has been on the BDD100k dataset [12], where we compared our network with MultiNet and DLT-Net [19], which are capable of handling multiple panoptic driving perception [20] tasks with impressive results in object detection [17] and drivable area segmentation [9]. We also evaluated Faster-R-CNN, YOLOv8 and PSPNet [20], which have demonstrated outstanding performance in object detection [9], semantic segmentation [9] and aggregating global information. We retrained these networks on the BDD100k dataset [12] for comparison with our network. We compared our network(YOLOv8) against ENET, SCNN and ENET-SAD for lane detection [8], as there are currently no multi-task networks that handle lane detection jobs on the BDD100K [19] dataset, which are sophisticated lane detection [6] networks. To ensure a fair comparison, we resized images in the BDD100k dataset [6] from  $1280 \times 720 \times 3$  to  $640 \times 384 \times 3$  and during all control studies, the same experimental parameters and assessment criteria were used. Additionally, we compared the performance of joint training paradigms [9] the accuracy and speed of our multi-task model taught to handle many tasks with the one trained to execute a given task, while switching training paradigms of various sorts, also the performance of semantic segmentation [6] was performed using tasks combined with single-stage and two-stage detection tasks [2]. Finally, all experiments were run on Nvidia DGX (Figs. 3, 4 and 5).



**Fig. 3** Results of lane and line detection



**Fig. 4** Object detection using YOLOv8



**Fig. 5** Final results of lane, line and distance computation

## 4 Results

### 4.1 Lane and Line Detection

#### 4.1.1 Vehicle Detection

See Tables 1, 2 and 3.

## 5 Conclusion

Our proposed approach is capable of handling four driving perception tasks simultaneously, object detection [10], drivable area segmentation [8], and lane detection [11] and distance computation [24]. We have achieved remarkable results on the

**Table 1** Object detection (traffic) results: comparing the proposed YOLOv8 with state-of-the-art models

Network	Recall (%)	mAP50 (%)	Speed (fps)
MultiNet	81.6	61.2	8.1
DLT-Net	85.6	67.4	9.1
Faster R-CNN	74.2	59.6	9.3
YOLOv5s	85.8	77.8	42
YOLOv8 (our model)	91.4	84.6	54

**Table 2** Drivable area segmentation results: comparing the proposed model with state-of-the-art drivable area segmentation semantic segmentation methods

Network	mIoU (%)	Speed (fps)
MultiNet	70.6	8.4
DLT-Net	72.3	9.5
Faster R-CNN	71.2	11.2
YOLOv5s	73.6	41
YOLOv8 (our model)	88	52

**Table 3** Lane and line detection results: comparing the proposed model with state-of-the-art lane and line detection methods

Network	mIoU (%)	Speed (fps)	Accuracy (%)
ENet	14.64	100	34.12
SCNN	15.84	19.8	35.8
ENet-SAD	16.02	36.56	41.3
YOLOv8 (our model)	26.20	78.63	75.8

challenging BDD100k dataset, surpassing state-of-the-art performance [3] on each of the four tasks. Our grid-based prediction [10] method is strongly connected to semantic segmentation [13], which is crucial for future research on multi-task learning [9]. Although the performance of our existing model is unaffected by end-to-end training, we intend to enhance task performance by utilizing a better suited multi-task learning[11] paradigm. In addition, our model is limited to three tasks, but future research iterations may include more autonomous perception [6] in driving, tasks such as depth estimation [6] and many more.

## References

1. Dihua W, Lv S, Jiang M, Song H (2020) Using channel pruning-based yolo v4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments. *Comput Electron Agric* 178:105742
2. Fang W, Wang L, Ren P (2020) Tinier-yolo: a real-time object detection method for constrained environments. *IEEE Access* 8:1935–1944
3. Girshick R (2015) Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision (ICCV)
4. Gkioxari G, Malik J, Johnson J (2019) Mesh r-cnn. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV)
5. He K, Gkioxari G, Dollar P, Girshick R (2017) Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision (ICCV)
6. Hu J, Sun Y, Xiong S (2021) Research on the cascade vehicle detection method based on CNN. *Electronics* 10(4)
7. Huang R, Pedoeem J, Chen C (2018) Yolo-lite: a real-time object detection algorithm optimized for non-GPU computers. In: 2018 IEEE international conference on big data (Big Data), pp 2503–2510
8. Huang Z, Wang J, Xuesong F, Tao Yu, Guo Y, Wang R (2020) DC-SPP-YOLO: dense connection and spatial pyramid pooling based YOLO for object detection. *Inf Sci* 522:241–258
9. Jiang H, Learned-Miller E (2017) Face detection with the faster r-cnn. In: 2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017), pp 650–657
10. Li C, Li L, Jiang H, Weng K, Geng Y, Li L, Ke Z, Li Q, Cheng M, Nie W, Li Y, Zhang B, Liang Y, Zhou L, Xu X, Chu X, Wei X, Wei X (2022) Yolov6: a single-stage object detection framework for industrial applications
11. Juan D (2018) Understanding of object detection based on CNN family and yolo. *J Phys Conf Ser* 1004(1):012029
12. Li Z, Pan M, Zhang T, Li X (2021) Testing dnn-based autonomous driving systems under critical environmental conditions. In: Meila M, Zhang T (eds) Proceedings of the 38th International conference on machine learning, vol 139 of proceedings of machine learning research. PMLR, 18–24 July 2021, pp 6471–6482
13. Liu C, Tao Y, Liang J, Li K, Chen Y (2018) Object detection based on yolo network. In: 2018 IEEE 4th information technology and mechatronics engineering conference (ITOEC), pp 799–803
14. Long X, Deng K, Wang G, Zhang Y, Dang Q, Gao Y, Shen H, Ren J, Han S, Ding E, Wen S (2020) PP-YOLO: an effective and efficient implementation of object detector
15. Lu X, Li B, Yue Y, Li Q, Yan J (2019) Grid r-cnn. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
16. Nguyen DT, Nguyen TN, Kim H, Lee HJ (2019) A high-throughput and power-efficient FPGA implementation of YOLO CNN for object detection. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 27(8):1861–1873
17. Okuyama T, Gonsalves T, Upadhyay J (2018) Autonomous driving system based on deep q learnig. In: 2018 International conference on intelligent autonomous systems (ICoIAS), pp 201–205
18. Ren S, He K, Girshick R, Sun J (2015) Faster r-cnn: towards real-time object detection with region proposal networks. In: Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R (eds) Advances in neural information processing systems, vol 28. Curran Associates, Inc
19. Shafiee MJ, Chywl B, Li F, Wong A (2017) Fast YOLO: a fast you only look once system for real-time embedded object detection in video
20. Tao J, Wang H, Zhang X, Li X, Yang H (2017) An object detection system based on YOLO in traffic scene. In: 2017 6th International conference on computer science and network technology (ICCSNT), pp 315–319
21. Wang CY, Bochkovskiy A, Liao HYM (2022) Yolov7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors

22. Wu D, Liao MW, Zhang WT, Wang XG, Bai X, Cheng WQ, Liu W (2021) YOLOp: you only look once for panoptic driving perception
23. Wu W, Liu H, Li L, Long Y, Wang X, Wang Z, Li J, Chang Y (2021) Application of local fully convolutional neural network combined with YOLO v5 algorithm in small target detection of remote sensing image. *PLOS ONE* 16(10):1–15, 10
24. Zhao L, Li S (2020) Object detection algorithm based on improved YOLOv3. *Electronics* 9(3)

# Schistosomiasis Prevention Undermined by Water Point Forecasting: An Approach Based on Data Fusion Model



Teegwende Zougmore, Sadouanouan Malo, and Bamba Gueye

**Abstract** In this study, we present an approach called FuMalMMo for the Fusion of Machine learning and Mathematical Models. It is an approach based on data fusion which leverages information coming from different data sources to make a decision. The approach we present follows a “Y” pattern where in the left branch there is a machine learning model in charge of forecasting the water quality of a water point. In the right branch, there is an epidemiological model responsible for making a forecast of the evolution of the density of parasites and snails causing schistosomiasis. In the middle branch, we rely on the theory of belief functions or evidence theory to combine the forecasts made by the two models in order to infer one day ahead the state of infestation of a water point with an accuracy of 0.75.

**Keywords** Data fusion · Water quality prediction · Mathematical epidemiology of infectious diseases · Schistosomiasis

## 1 Introduction

Schistosomiasis is an acute and chronic parasitic disease caused by Trematoda of the genus *Schistosoma*. The larvae of the parasite, released by intermediate hosts (snails), enter a person’s skin when in contact with infested water. The life cycle of the disease transmission involves humans (final host), parasites, and intermediate hosts (snails). The intermediate hosts (snails) live in water points (lakes, dams, rivers, etc.). The

---

T. Zougmore (✉)

Université Alioune Diop de Bambey, Bambey, Senegal  
e-mail: [teeg-wende@uadb.edu.sn](mailto:teeg-wende@uadb.edu.sn)

S. Malo

Université Nazi Boni, Bobo-Dioulasso, Burkina Faso  
e-mail: [sadouanouan@yahoo.fr](mailto:sadouanouan@yahoo.fr)

B. Gueye

Université Cheikh Anta Diop, Dakar, Senegal  
e-mail: [bamba.gueye@ucad.edu.sn](mailto:bamba.gueye@ucad.edu.sn)

water quality of these water points influences their biological cycle as well as that of the parasites [1–3], and consequently plays on the extinction or the persistence of the disease.

We address the spread of schistosomiasis from a health prevention viewpoint. Indeed, we aim to set up an alert system to prevent when a water point is infested. A water point is infested when it contains infected snails that emit parasites. To achieve this, it is necessary to evaluate earlier the quality of water points and the density evolution of snails and parasites. For the earlier assessment of water quality, we consider a water quality model which can forecast one day ahead the quality of a water point [4]. And for the assessment of the snail's and parasite's density evolution, we refer to an epidemiological model. To leverage the two sources of information, we formulate an approach which consists of fusion of the results of the two mentioned models.

In [5] data fusion is defined as a combination of information originating from several sources in order to improve decision-making. Several methods of data fusion are encountered in the literature. They come essentially from probability theory, evidence theory, and fuzzy set theory [6–8]. The methods are proposed to take into account the characteristics of the information to be combined. The authors in [9] indicate that these different methods are not to be put in competition and are not contradictory. They argue that the choice of one of these methods must be made by finding the best match between the intended application and the specifics of the method.

The information to be fused in our case study presents a form of imperfection which is uncertainty. This uncertainty is due to the fact that none of the forecasting models used as a data source is intended to faithfully reflect reality. This results in forecasts with margins of error. Uncertainty is represented and quantified by probability theory [6, 8]. But its use requires a priori probabilities [8]. These a priori probabilities are difficult to determine in our case study. Evidence theory is a data fusion method that does not require knowledge of a priori probabilities [10]. It extends both set theory and probability theory in the representation of uncertainty [11]. This leads us to employ evidence theory in our study.

The rest of the paper is organized as follows: Sect. 2 presents some basic concepts of evidence theory. In Sect. 3, we describe the proposed data fusion architecture corresponding to our case study. Section 4 explains how evidence theory is applied to combine data coming from the data sources. We present the experimental setup and results respectively in Sects. 5 and 6. Section 7 gives a conclusion and some perspectives to be addressed in future work.

## 2 Background on Evidence Theory

The theory of belief functions, also known as the Dempster-Shafer theory or the theory of evidence, was proposed by Dempster and then mathematically formulated by Shafer [8, 9]. It is based on modeling the belief in an event. There are four steps

to follow in the application of this theory. The first step is modeling which consists of choosing a mathematical representation for information to be combined. The second step consists of quantifying the information. The third and fourth steps are combination and decision which consist of applying rules to synthesize information and take decision.

## 2.1 Modeling

Information to be fused is modeled by mass function or basic belief assignment. By setting  $D = \{d_1, d_2, \dots, d_n\}$ , the frame of discernment where each  $d_i$  designates a hypothesis in favor of which a decision can be made, the mass function can be defined on  $2^D$  with values in  $[0, 1]$  [6, 12]. For a source  $S_j$ , the mass function  $m_j$  verifies:

$$\sum_{A \in 2^D} m_j(A) = 1 \quad (1)$$

From a mass function, it is possible to derive other functions [9, 13] such as: (i) the credibility denoted “bel” which represents the total mass of belief in A and the plausibility “Pl” which is interpreted as the maximum belief in A. We just present the equation of ‘Pl’ that we need in decision step. “Pl” is defined as follows:

$$Pl_j(A) = \sum_{A \cap B} m_j(B) \forall A \subseteq D \quad (2)$$

Mass function can be discounted to take into account the reliability of sources. It is done by introducing a discounting coefficient  $\alpha_j \in [0, 1]$  [13]. The mass function for all  $A \in 2^D$ ,  $A \neq D$  is thus redefined as:

$$\begin{cases} m_j'(A) = (1 - \alpha_j)m_j(A), A \subset D \\ m_j'(D) = (1 - \alpha_j)m_j(D) + \alpha_j \end{cases} \quad (3)$$

## 2.2 Estimation

Except when an expert expresses his opinion in the form of a mass function directly, in all other cases, there is no generic method to solve this problem [9]. We expose here a mass function deduced from a probability of realization  $s \in [0, 1]$  of the hypothesis or set of hypotheses A. Thus, we have:

$$\begin{cases} m_j(A) = s, A \subset D \\ m_j(\bar{A}) = 1 - s \\ m_j(B) = 0, B \neq A \subset D \end{cases} \quad (4)$$

where  $s \in [0, 1]$ , a real is considered as the probability of occurrence of the event  $A$ .

### 2.3 Combination

We present here the Dempster-shafer's rule which is the rule that we have used. It operates a conjunctive combination followed by normalization. The normalization consists of distributing the mass of the conflict to all the other elements of  $2^D$  except  $\emptyset$ . We describe this rule by considering only two mass functions,  $m_1$  and  $m_2$ . We propose to distinguish the combined mass function by the notation  $m_{comb}$ . The combination performed by the Dempster-Shafer's rule is defined as follows:

$$m_{comb} = (m_1 \oplus m_2)(A) = \frac{1}{1 - k} \sum_{B_1 \cap B_2 = A} m_1(B_1)m_2(B_2) \quad (5)$$

With  $m_{comb}(\emptyset) = 0$  and  $k$  the normalization term is:

$$k = \sum_{B_1 \cap B_2 = \emptyset} m_1(B_1)m_2(B_2) \quad (6)$$

### 2.4 Decision

The choice of the final decision or hypothesis can be made according to several criteria [6, 13]. The criterion used to determine the final decision in our context is maximum plausibility. Let  $Dec$  denote this decision and  $Pl_{comb}$  the plausibility deduced from the combined mass  $m_{comb}$ .  $Dec$  can be defined as follows:

$$Dec = \operatorname{argmax} Pl_{comb}(A) \quad (7)$$

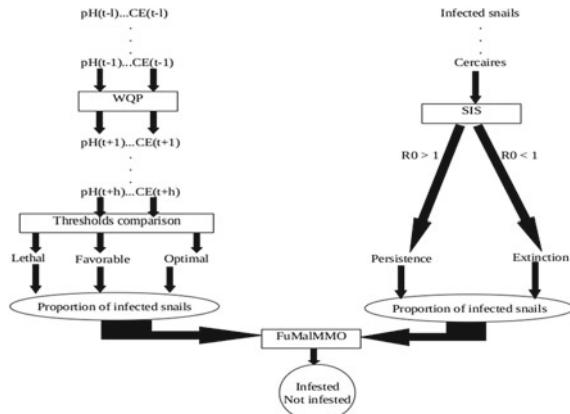
### 3 Proposed Data Fusion Architecture

The proposed architecture comes in three branches in the form of a "Y" as illustrated in Fig. 1. The left branch is responsible for providing information relating to the water quality of a water point. It is based on the water quality forecasting model that we have developed in a previous work [4]. We denote this model WQP for Water Quality Prediction. The model forecasts one day ahead the future values of pH, temperature, dissolved oxygen (DO), and electrical conductivity (EC). Then these predicted values are after compared with thresholds taken from the literature to characterize the water quality as lethal, favorable, and optimal. Each category of water quality corresponds to a specific proportion of infected snails. We show in Sect. 4.1 how it is determined.

The right branch is responsible for providing information relating to the evolution of the density of snails and parasites. It is based on an epidemiological model that we have identified in a previous work [14]. We denote this model SIS for Susceptible Infected Susceptible. It is a mathematical model with compartments which take as input the number of human populations, parasites (miracidia and cercariae), and intermediate hosts (snails) and provide the basic reproduction number  $R_0$ .  $R_0$  governs the dynamics of the system. If  $R_0 > 1$ , there is a proliferation of infected snails and the disease persists. If  $R_0 < 1$ , there is an extinction of the disease. Each condition met by the reproductive number value corresponds to a specific proportion of snails also infected. We show in Sect. 4.1 how it is determined.

The proportions of snails determined according to the output of each model will constitute the data to be combined. The middle branch is responsible for this combination truly speaking itself. This involves performing the four stages of the evidence theory in order to infer the infestation state of a water point. We denote it FUMalMMO for the Fusion of Machine learning Model and Mathematical Model.

**Fig. 1** Data fusion architecture



## 4 Implementation of Evidence Theory

### 4.1 Mass Modeling and Estimation

The water point will be declared infested or not. This leads us to define a frame of discernment which is made up of two candidate hypotheses:  $D = \{d_1, d_2\}$ .  $d_1$  is the hypothesis that an observed water point is infested.  $d_2$  is the assumption that it is not. The power set is then  $2^D = \{\emptyset, d_1, d_2, d_{1,2}\}$ .  $d_{1,2} = \{d_1, d_2\}$  and allows translating a part of ignorance on the state of the water point.

**Mass function modeling.** For each possible output of WQP, the defined mass function is  $m_{wqp}$ :

$$\{m_{wqp}(d_1), m_{wqp}(d_2), m_{wqp}(d_{1,2})\} \quad (8)$$

With  $m_{wqp}(d_1)$  being a numeric value indicating how much WQP believes the water is infested.  $m_{wqp}(d_2)$  indicates how much WQP believes that the water point is not infested.  $m_{wqp}(d_{1,2})$  allows WQP to express ignorance as to the probable state of the water point. And for each output of SIS, the defined mass function is  $m_{sis}$ :

$$\{m_{sis}(d_1), m_{sis}(d_2), m_{sis}(d_{1,2})\} \quad (9)$$

$\{m_{sis}(d_1), m_{sis}(d_2), m_{sis}(d_{1,2})\}$  have the same meaning as  $\{m_{wqp}(d_1), m_{wqp}(d_2), m_{wqp}(d_{1,2})\}$  but from the point of view of SIS.  $m_{wqp}(\emptyset) = 0$  and  $m_{sis}(\emptyset) = 0$  because the discernment framework contains all possible candidate hypotheses. Once the choice of representation has been made, the next step is estimation.

**Estimation of mass functions.** We consider the Eq. 4 described in Sect. 2.2 to estimate the mass functions. The probability of realization of the hypotheses will correspond to a proportion of infected snails. This proportion depends on the different outputs of WQP and SIS models as described in the architecture in the Sect. 3. For each output of the WQP model, the mass function is estimated as follows:

$$\begin{cases} m_{wqp}(d_1) = t, d_1 \subset D \\ m_{wqp}(d_2) = 1 - t \end{cases} \quad (10)$$

For each output of the SIS model, the mass function is estimated as follows:

$$\begin{cases} m_{sis}(d_1) = t, d_1 \subset D \\ m_{sis}(d_2) = 1 - t \end{cases} \quad (11)$$

where  $t$  denotes the proportion of infected snails. In the following lines, we indicate how it is determined from each model.

**Table 1** Estimated mass functions for each output of WQP

Water quality category	$m_{wqp}(d_1)$	$m_{wqp}(d_2)$	$m_{wqp}(d_{1,2})$
Lethal	0	1	0
Favorable	0.32	0.68	0
Optimal	0.64	0.36	0

*Determination of the proportion of infected snails from WQP.* It is carried out on the basis of spatio-temporal malacological study. That is to say, a study comprising a collection of snails carried out at a given period in different places and followed by an identification of the species and then a test for the emission of cercariae. To all this is added an analysis of the physicochemical parameters to establish a correlation with the proportion of infected snails. The infected snails' proportion is determined by the following formula:

$$t^q = \sum_{i=1}^n P_i^q \quad (12)$$

where  $t^q$  indicates the arithmetic mean of infected snails proportion corresponding to water quality  $q$ .  $q$  being a flag which designates one of the possible outputs of WQP: lethal, favorable, and optimal.  $P_i^q$  indicates the infected snail's proportion found on a specific place  $i$  and corresponding to water quality  $q$ .

We selected a study which took place in Tanzania (at Lake Victoria) and which focused on snails belonging to the genus *Biomphalaria* which is the genus responsible for intestinal schistosomiasis [15]. The study took place from February 2016 to March 2016. Sixteen places were explored. For each category of water quality, we calculate the arithmetic mean of the proportions. Thus, we obtain  $t^{optimal} = 63.68\%$  for the optimal category and  $t^{favorable} = 32.43\%$  for the favorable category and  $t^{lethal} = 0\%$ .

Once the proportion has been determined for each category, the resulting mass functions are deduced by Eq. 4. We present the results in Table 1.

The values in Table 1 are interpreted as follows: when water is lethal, a mass function of 1 is assigned to the hypothesis  $d_2$ . This is interpreted as follows: when the water is lethal, there can be no infestation. When the water is favorable, a mass function of 0.3243 is assigned to the hypothesis  $d_1$  and a mass function of 0.6757 is assigned to the hypothesis  $d_2$ . This is interpreted as follows: when the water is favorable, the chances that the point is infested are very low, but it is not excluded that it is not. When water is optimal, a mass function of 0.64 is assigned to the hypothesis  $d_1$ , and a mass function of 0.36 is assigned to the hypothesis  $d_2$ . This is interpreted as follows: when the water is optimal, the chances that the water point is infested are high but nothing excludes that it is not.

*Determination of the proportion from SIS.* The proportion of infected snails is determined based on the value of the basic reproduction number  $R_0$ . When:

- $R_0 < 1$ , the disease will die out. This means that the system is in a state where the infected compartments do not have enough individuals for the disease to

**Table 2** Estimated mass functions for each output of SIS

$R_0$ threshold	$m_{sis}(d_1)$	$m_{sis}(d_2)$	$m_{sis}(d_{1,2})$
$R_0 < 1$	0	1	0
$R_0 > 1$	0.85	0.15	0

spread. This state is translated by the solution of the system which is  $\varepsilon^0 = (H0, 0, 0, M, 0, 0)$ . This means that the compartments of susceptible humans and snails have individuals. But there are neither parasites, nor infected individuals. From this, we derive a proportion of infected snails  $t = 0\%$ .

- $R_0 > 1$ , the disease will spread. This means that the system is in a state where there are enough individuals in the infected compartments to allow disease transmission. This state is translated by a solution of the system which is  $\varepsilon^* = (HS^*, HI^*, K^*, MS^*, MI^*, P^*)$ . This solution is not an explicit expression like  $\varepsilon^0$ . To determine a proportion of infected snails, it is necessary to:

- launch a numerical simulation of the system;
- and calculate a proportion when the variations of the different compartments become insignificant. The calculation is done with the following formula:

$$t = MI^*/(MS^* + MI^*) \quad (13)$$

With  $(MS^* + MI^*)$  representing the total number of snails observed and  $MI^*$  the number of snails emitting cercariae. For the numerical resolution, we use the simulation data of the model found in [16]. We thus obtain  $t = 85\%$ .

Once the proportion has been determined for each value of  $R_0$ , the resulting mass functions are deduced by Eq. 4. We present the results in Table 2.

The values in the Table 1 are interpreted as follows: if  $R_0 < 1$ , a belief mass of 1 is estimated for the hypothesis  $d_2$ . This results in the fact that there cannot be infestation when there is extinction of the disease. If  $R_0 > 1$ , a mass of 0.85 is estimated for the hypothesis  $d_1$  and a mass of 0.15 for the hypothesis  $d_2$ . This translates into the fact that when the disease persists, there is a very good chance that the water point will be infested. But nothing excludes that this is not the case.

**Discounting of mass functions.** Our sources are forecast models. This indicates that there may be discrepancies between the predicted and expected values. To take into account the errors of these models, we propose to determine a discounting coefficient  $\alpha$  by the following formula:

$$\alpha = 1 - R^2 \quad (14)$$

With  $R^2$  denoting the coefficient of determination. We therefore obtain for the source WQP:

$$\alpha_{wqp} = 1 - R_{wqp}^2 \quad (15)$$

and for source SIS:

$$\alpha_{sis} = 1 - R_{sis}^2 \quad (16)$$

Once the discounting coefficients are determined, the Eq. 3 is applied. The masses of belief become thus:

$$\begin{cases} m'_{wqp}(d_1) = (1 - \alpha_{wqp})m_{wqp}(d_1) \\ m'_{wqp}(d_2) = (1 - \alpha_{wqp})m_{wqp}(d_2) \\ m'_{wqp}(d_{1,2}) = (1 - \alpha_{wqp})m_{wqp}(d_{1,2}) + \alpha_{wqp} \end{cases} \quad (17)$$

And

$$\begin{cases} m'_{sis}(d_1) = (1 - \alpha_{sis})m_{sis}(d_1) \\ m'_{sis}(d_2) = (1 - \alpha_{sis})m_{sis}(d_2) \\ m'_{sis}(d_{1,2}) = (1 - \alpha_{sis})m_{sis}(d_{1,2}) + \alpha_{sis} \end{cases} \quad (18)$$

We use the symbol “ $\gamma$ ” to designate the discounted masses.

## 4.2 Combination of Mass Functions

The Dempster-Shafer combination rule described by the Eq. 5 is applied to obtain the mass set overall after the merger. The normalization term k obtained is equal to:

$$k = m'_{wqp}(d_1)m'_{sis}(d_2) + m'_{sis}(d_1)m'_{wqp}(d_2) \quad (19)$$

We thus obtain the combined masses of belief:

$$\begin{cases} m_{comb}(d_1) = \frac{1}{1-k}[m'_{wqp}(d_1)m'_{sis}(d_1) + m'_{wqp}(d_1)m'_{wqp}(d_{1,2}) + m'_{sis}(d_1)m'_{sis}(d_{1,2})] \\ m_{comb}(d_2) = \frac{1}{1-k}[m'_{wqp}(d_2)m'_{sis}(d_2) + m'_{wqp}(d_2)m'_{wqp}(d_{1,2}) + m'_{sis}(d_2)m'_{sis}(d_{1,2})] \\ m_{comb}(d_{1,2}) = \frac{1}{1-k}[m'_{wqp}(d_{1,2})m'_{sis}(d_{1,2})] \end{cases} \quad (20)$$

### Decision of mass function

At this stage, there is a calculation of plausibilities which is carried out with Eq. 2. We obtain:

$$\begin{cases} Pl_{comb}(d_1) = m_{comb}(d_1) + m_{comb}(d_{1,2}) \\ Pl_{comb}(d_2) = m_{comb}(d_2) + m_{comb}(d_{1,2}) \\ Pl_{comb}(d_{1,2}) = m_{comb}(d_1) + m_{comb}(d_2) + m_{comb}(d_{1,2}) \end{cases} \quad (21)$$

Then we retain the hypothesis with the maximum plausibility as the decision to be made.

$$Dec = \operatorname{argmax}(Pl_{comb}(d_1), Pl_{comb}(d_2)) \quad (22)$$

In the final result, the water point has to be forecasted as “infested” or “not infested”, i.e. there is no reject. Consequently, the final decision is “infested” if the plausibility of the hypothesis  $d_1$  is greater than that which corresponds to  $d_2$  whatever the plausibility of  $d_{1,2}$ .

## 5 Evaluation

### 5.1 Experimental Setup

**Required data.** Carrying out the experiment requires data which can serve as inputs for WQP model and SIS model.

A study providing this necessary data was conducted in [17]. It was carried out from January 2016 to May 2017 in Panamasso and focused, among other things, on a parasitological study and a malacological study associated with an analysis of the physicochemical parameters of a water point.

The parasitological study took place precisely in January 2016 and revealed that the human prevalence rate of intestinal schistosomiasis is 27.47% and the total size of the population is 3065 inhabitants. The malacological study consisted of collecting snails of the genus *Biomphalaria* (responsible for intestinal schistosomiasis) and measuring physicochemical parameters such as PH, temperature (TEMP), electrical conductivity (EC), and dissolved oxygen (DO). The different collections took place in three seasons: winter (June 2016–November 2016), cold (December 2016–February 2017) and hot (March 2017–May 2017). Each collection phase was spread over 10 days.

For our test, we are interested in two seasons, the cold season and the hot season. We summarize the data of these two seasons in Table 3. In addition to the physico-chemical parameters, the density of snails collected and the proportion of infected snails are indicated.

In addition to these data relating to the specific case of the Panamasso water point, other data relating to “biological” and “contextual” parameters are needed. For the first type, the values are taken from the literature [16]. For the contextual parameters, we used the 2016 statistical yearbook of Burkina Faso [18]. These are the human natural birth rate (46/1000) and the life expectancy (56.7 years) to determine the recruitment rate and the human natural death rate respectively.

**Table 3** Average of physicochemical parameters and malacological data

Season (collection month)	Parameter	Value	Density (/30 min)	Proportion of infected snails (%)
Cold (December 2016)	PH	7	64.62	40.04
	TEMP (°C)	24		
	CE	125		
	DO (%)	75		
Hot (March–April 2017)	PH	6	187.66	59.18
	TEMP (°C)	27		
	CE	220		
	DO (%)	35		

**Testing process.** We proceed by:

- A definition of some scenarios from the data presented in Table 3 and by simulating the application of certain control methods that break the cycle of transmission of the disease. The different scenarios established are presented in Table 4.
- An execution of WQP and SIS models as follows:
  - The SIS model takes as input the December population densities and the necessary parameters; it then provides a  $R_0$  valid for the period from January 2017 to March 2017;
  - The WQP model takes as input all the physicochemical parameters of the past two days; we start with the last two days of December 2016. Then the last day of December 2016, the 1st day of January 2017, and so on until we cover the entire period from January 2017 to March 2017. For each observation of the two last days, it provides the next day's water quality;
  - An execution of the *FuMalMMO* model which consists of applying the evidence theory as described in Sect. 4 on the forecasts of water quality and the evolution of the density of snails. We then randomly choose six dates in the period from January 2017 to March 2017. And we perform the fusion taking into account the value of  $R_0$  of the period and the water quality of the water point forecasted for each of these dates.

**Table 4** Scenarios determined on the basis of physicochemical parameters and malacological and parasitological data

Scenarios	Control method	Expected situation
1	None of human recovers	Infested Water
2	Full recovery of illness humans (100%)	Infested Water
3	Removal of mollusks via molluscide	Uninfested water
4	Environmental Pollution	Uninfested water

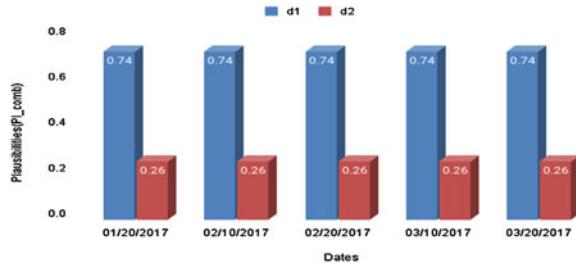
## 6 Results

### 6.1 Infestation States Observed for Each Scenario

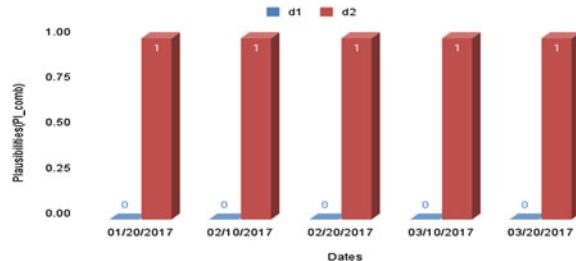
The basic reproduction numbers obtained for the different scenarios are as follows:  $R_0 = 39$  for scenario 1 and scenario 4.  $R_0 = 0$  for scenario 2 and scenario 3. We present the results obtained for the different scenarios in Figs. 2, 3, 4, 5.

For scenario 1, all the chosen dates indicate a situation of an infested water point. This result is consistent with the expected situation as mentioned in Table 4. Scenarios are determined on the basis of physicochemical parameters and malacological and parasitological data. For scenario 2, the chosen dates indicate a situation of an uninfested water point. These results are contrary to the expected situation mentioned in Table 4. For scenario 3, all the chosen dates indicate a situation of an uninfested

**Fig. 2** Scenario 1

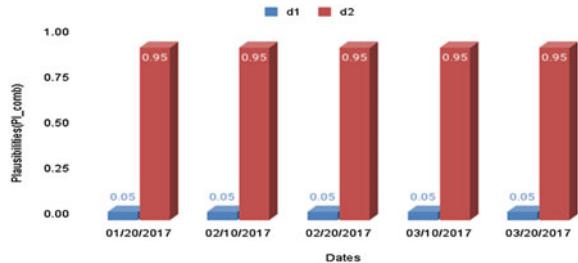


**Fig. 3** Scenario 2



**Fig. 4** Scenario 3



**Fig. 5** Scenario 4

water point. But at this time, the results are consistent with the expected situation. For scenario 4, we obtain a situation of an uninfested water point for the chosen dates. These results are consistent with the expected situation mentioned in Table 4.

These results are obtained from the quality of the water point forecasted on the chosen dates and the value of  $R_0$  of the period from January 2017 to March 2017. In Tables 5, 6, 7, and 8, it is presented the different operations performed to infer the infestation state of water on a chosen date. The date 2017-03-10 is taken as an example. The same operations are done for the other chosen dates. The column headers of the tables indicate the different candidate hypotheses. The row headers represent respectively the mass functions assigned by WQP and SIS as well as the combination and decision operations. The contents of the tables are different according to the mass functions assigned following the prediction results of the two models. In scenario 1, the water quality forecast is favorable and the basic reproduction number  $R_0$  is equal to 39. In scenario 2, the water quality forecast is favorable and the basic reproduction number  $R_0$  is equal to 0. In scenario 3, the water quality forecast is favorable and the basic reproduction number  $R_0$  is equal to 0. And in scenario 4, the water quality forecast is lethal and the basic reproduction number  $R_0$  is equal to 39.

**Table 5** Mass functions obtained on date of 2017-03-10 for scenario 1

	$d_1$	$d_2$	$d_{1,2}$
$m_{wqp}$	0.32	0 67	0.01
$m_{sis}$	0.85	0.1	0
$m_{comb}$	0.74	0.26	0
$Pl_{comb}$	0.74	0.26	1

**Table 6** Mass functions obtained on date 2017-03-10 for scenario 2

	$d_1$	$d_2$	$d_{1,2}$
$m_{wqp}$	0.32	0.67	0.01
$m_{sis}$	0	1	0
$m_{comb}$	0	1	0
$Pl_{comb}$	0	1	1

**Table 7** Mass functions obtained on date of 2017-03-10 for scenario 3

	$d_1$	$d_2$	$d_{1,2}$
$m_{wqp}$	0 32	0.67	0.01
$m_{sis}$	0	1	0
$m_{comb}$	0	1	0
$Pl_{comb}$	0	1	1

**Table 8** Mass functions obtained on date 2017-03-10 for scenario 4

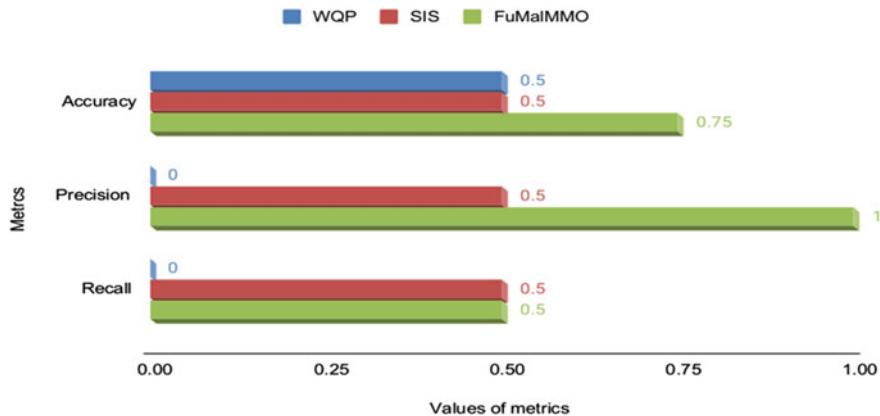
	$d_1$	$d_2$	$d_{1,2}$
$m_{wqp}$	0	0.99	0.01
$m_{sis}$	0.85	0.15	0
$m_{comb}$	0.05	0.95	0
$Pl_{comb}$	0.05	0.95	1

## 6.2 Overall Performance of the FuMalMMO Model

The FuMalMMO model relies on two forecasting models to infer the infestation status of a water point. It thus behaves like a binary classification model. We evaluate here its performance in relation to the different scenarios defined in order to assess its ability to infer in advance the state of infestation of a water point. To do this, we use different measures that are used to quantitatively assess the obtained results. The water point studied is either in an infested state or not. We call the infested state positive and the uninfested state negative. A correctly classified state is considered true positive or true negative. On the contrary, a misclassified condition is either a false positive or a false negative. True positives and false positives resulting from our method are compared against expected situations. To assess performance, we adopt the following metrics: accuracy, precision, and recall. We do not go further on how these metrics are calculated. One can refer to [19] for more information.

In Fig. 6, the overall performance according to these three metrics is illustrated. We also calculated these metrics for each of the models involved in the fusion taken separately. To do this, it was considered that each model taken individually should decide on its own, the state of the water point based on the hypothesis having received the greatest plausibility among those it has issued.

The x-axis of the graph in Fig. 6 represents the values of the metrics. On the y-axis, we have the different metrics used for the evaluation of the models. With the data fusion approach of the two WQP and SIS models, we end up with a FuMalMMO model which is able to classify in advance with great confidence the cases of infestation (Precision = 1). On the other hand, when it comes to classifying all the positive cases in advance, its performance is average (recall = 0.5). But we note that it is able to classify in advance all types of cases with good accuracy (0.75). By comparing



**Fig. 6** Comparison of *FuMalMMo* against *WQP* and *SIS*

the values of the metrics of FuMalMMO with those obtained from WQP and SIS, we can note that each model taken separately has modest performances compared to those of the fusion model which combines the two.

## 7 Conclusion

Data fusion makes it possible to leverage several pieces of information in order to make a better decision. In this paper, we have shown how to use data fusion to assess earlier if a water point harbors infected snails. The fusion method used is the evidence theory or belief functions theory.

Using this theory, we achieved the fusion of information from two complementary models. One is a machine learning model forecasting water quality and the other is an epidemiological model forecasting the density evolution of snails and parasites.

With a recall of 0.5, an accuracy of 0.75, and a precision of 1, the fusion approach leads to an efficient model to warn one day ahead that a water point is likely to be infested with parasites causing schistosomiasis.

In this work, the mass functions assigned by the water quality prediction model are fixed. We think it would be interesting to consider determining them dynamically. That is to say, instead of categorizing the water quality into lethal, favorable, and optimal and then determining the belief masses afterward, we will try to determine these mass functions via a regression model.

## References

1. Abdel Malek E (1958) Factors conditioning the habitat of bilharziasis intermediate hosts of the family Planorbidae. Bull World Health Organ 18(5–6):785–818
2. Félicien GY, Rose DN, Faustin AO, Eliézer K, Dynamique des populations de mollusques hôtes intermédiaires de *Schistosoma haematobium* et *Schistosoma mansoni* dans le lac du barrage de Taabo (sud Côte d'Ivoire) 3:15
3. Bakhoum S et al (2019) Influence des paramètres physico-chimiques sur la répartition spatiale des mollusques hôtes intermédiaires des schistosomes humains dans le delta du fleuve Sénégal. Médecine Santé Trop 29(1):61–67. <https://doi.org/10.1684/mst.2019.0883>
4. Zougmore T, Gueye B, Malo S (2022) An AI-based approach to the prediction of water points quality indicators for schistosomiasis prevention. In: 2022 IEEE multi-conference on natural and engineering sciences for Sahel's sustainable development (MNE3SD), pp 1–6. <https://doi.org/10.1109/MNE3SD53781.2022.9723395>
5. B (I ), MAITRE (H.) (1994) 01 - Fusion de données en traitement d'images: modèles d'information et décisions, vol. 11. GRETSI, Saint Martin d'Hères, France
6. Bloch I (2005) Fusion d'informations numériques: panorama méthodologique. J Natl Rech En Robot 2005:79–88
7. Khaleghi B, Khamis A, Karray FO, Razavi SN (2013) Multisensor data fusion: a review of the state-of-the-art. Inf Fusion 14(1):28–44. <https://doi.org/10.1016/j.inffus.2011.08.001>
8. Chahine C (2016) Fusion d'informations par la théorie de l'évidence pour la segmentation d'images. PhD Thesis, Université Paris-Est; École Doctorale des Sciences et de Technologie (Beyrouth)
9. Lefèvre E (2012) Fonctions de croyance: de la théorie à la pratique. PhD Thesis
10. Bouain M (2019) Système embarqué de fusion multi-capteurs pour la détection et le suivi d'obstacles statiques et dynamiques. PhD Thesis, Université de Valenciennes et du Hainaut-Cambrésis
11. Denœux T (2010) Théorie des fonctions de croyance: application en reconnaissance de formes et en fusion d'informations. Journée «Théorie et applications des fonctions de croyance pour les systèmes ...
12. Martin A, Polycopié de cours ENSIETA - Réf. : 1484, p 117
13. Florea MC, Jousselme A-L, Bossé É, Grenier D (2009) Robust combination rules for evidence theory. Inf Fusion 10(2):183–197. <https://doi.org/10.1016/j.inffus.2008.08.007>
14. Zougmore T, Malo S, Gueye B, Ouaro S (2020) Toward a data fusion based framework to predict schistosomiasis infection. In: 2020 IEEE 2nd international conference on smart cities and communities (SCCIC), pp 1–8. <https://doi.org/10.1109/SCCIC51516.2020.9377330>
15. Fuss A, Mazigo HD, Mueller A (2020) Malacological survey to identify transmission sites for intestinal schistosomiasis on Ijinga Island, Mwanza, north-western Tanzania. Acta Trop 203:105289. <https://doi.org/10.1016/j.actatropica.2019.105289>
16. Traoré B (2019) Modélisation mathématique et simulations numériques des maladies infectieuses: cas du paludisme et de la bilharziose. Theses, Nazi BONI
17. Compaore FR (2018) Analyse spatio-temporelle des facteurs environnementaux et socio-sanitaires favorables à la persistance des maladies liées à l'eau : cas de la schistosomiase au Burkina Faso. INSTITUT INTERNATIONAL D'INGENIERIE DE L'EAU DE L'ENVIRONNEMENT
18. Burkina F (2016) Institut national de la statistique et de la Demographie. Annu Stat L'économie Financ
19. Erickson BJ, Kitamura F (2021) Magician's corner: 9. Performance metrics for machine learning models. Radiol Artif Intell 3(3). Radiological Society of North America, p e200126

# A Comparative Study of Classification Algorithms for the Prediction of Communication Effectiveness



S. Suji and D. Abisha

**Abstract** Machine learning is an essential element of the growing field of data science. The performance of communication prediction is crucial to understanding and identifying a student's level. In this Research, we are determining each student's communication level. After the outcome, teachers can give them proper training to improve their level. By discovering this level, we could recover and train the student before getting a suitable placement. Many students must catch up every year because of a lack of proper advice and monitoring. A teacher can only monitor some students at a time if a system can help a teacher with the students, like which student needs which kind of help. Also, it'll be important and helpful for both preceptors and scholars. We compare their results and levels in various Machine Learning Algorithms like Support Vector Machine (SVM), Random Forest Classifier, Logistic Regression, Naive Bayes, and Decision Tree for accuracy. This exploration would be helpful for scholars and preceptors.

**Keywords** Machine learning · Prediction · Communication level · Algorithm · Accuracy

## 1 Introduction

Information must be sent properly and efficiently to be communicated. Each employer can legitimately demand excellent communication abilities for any job description, and today, most businesses use online communication rounds to evaluate a candidate's communication skills. It is advised that students use this site to rehearse before going to placement drives. This study offers a method for using an application to practice and anticipate one's skills. The creation of an effective and

---

S. Suji · D. Abisha (✉)  
Department of Computer Science and Engineering, National Engineering College, Kovilpatti,  
India  
e-mail: [abisha\\_cse@nec.edu.in](mailto:abisha_cse@nec.edu.in)

S. Suji  
e-mail: [1912030@nec.edu.in](mailto:1912030@nec.edu.in)

precise prediction system is the aim of this research. To forecast future events and performance for a firm, predictive analytics employs statistics and modeling techniques. Businesses that employ predictive analytics are able to identify risks and opportunities. By anticipating their communication level and making recommendations to help them communicate better, our goal is to help students grow through self-paced learning. Most businesses use online communication rounds to gauge employees' level of communication proficiency. Students should practice using the site in advance of placement drives. Students' major goal is to increase their knowledge through self-paced learning. Students are urged to work on their communication skills and provide suggestions for improvement in the portal prior to attending placement drives. The majority of businesses prefer to disclose their rank online. The capacity to effectively and efficiently convey knowledge to others is known as communication. The ability to communicate effectively can be useful in both professional and social contexts. Knowledge professionals need to be great communicators in a variety of media. This encompasses non-verbal, textual, and spoken communication. In a variety of businesses and work roles, effective communication skills are crucial during media interviews. To ensure that important workers are portrayed favorably to the media, significant training expenses are made. Passing the information from one thing to another is the definition of communication, a complex phenomenon. We have gathered participant-specific interviews in order to accomplish this. The data is then subjected to various analyses, and a predictive model is created to evaluate the consortium members' communication abilities. In order to identify weak students and help them overcome their communication challenges, we would like to anticipate (before the placement drive) student Level (0, 1, 2) in the communication test. Early level forecasts will assist teachers in identifying students who require communication support, as well as assist students in addressing their areas of weakness to achieve a good level in the placement test.

## 2 Related Works

A machine learning algorithm was suggested by Santhana Krishnan and Geetha [1] to demonstrate the most accurate and dependable system. Varma et al. [2] seek to base their assessments on all fundamental factors taken into account when establishing price. It was also suggested to obtain precise real-world assessments using Google Maps' real-time neighborhood information. A prediction system is suggested by Joy and Raj [3] so that an institution's academic planning for subsequent years might be aided. As data mining and machine learning advanced, many predictor models were developed by analyzing the student dataset from the past. The literature review on placement prediction models for graduate engineering students in their first year is presented in this study. In this study, Ying [4] compares the predicting error for the number of institutions accepting students between the support vector regression approach and the BP neural network. They also offer forecasting curves for the two methods. The support vector regression approach has a higher forecasting accuracy

than BP neural network, according to comparison results of forecasting inaccuracy for colleges recruiting students between the two methods. Bujang et al. [5] provide a thorough examination of methods for forecasting first-semester course grades. In this study, two modules are highlighted. We must first compare the accuracy performance utilizing course grade records from 1282 actual students.

According to Wei's [6] study of experimental samples based on 800 students, the suggested strategy can provide up to 79.1% accurate models. The findings of Lee et al. [7] indicate that most respondents have high AQ scores and that students are predominately interested in careers that appeal to their social personalities. Based on AQ, Social, and Entrepreneur scores, this model is anticipated to forecast future employment performance for UTHM Bachelor's degree candidates. To predict the likelihood that students will enroll in the same course in the following term, Gull et al. [8] suggest using logistic regression, linear discriminant analysis, K-nearest neighbors, classification and regression trees, Naive Gaussian Bayes, and support vector machines on historical data of student grades in one of the undergraduate courses. Our research demonstrates that the best method for accurately predicting how well students will perform on final examinations is linear discrimination analysis. 49 records out of a total of 54 were correctly predicted by the model, yielding a 90.74% accuracy rate. The Markov-modified model-based video prediction caching technique put out by Ju et al. in their [9] paper can be used when there aren't many historical user access data. Simulation studies demonstrate that the dynamic prediction's realization increases hit rates and response times while confirming the algorithm's efficiency, accuracy, and quickness. By examining how the two factors affect the effectiveness of the LS-SVM classifier, Qiu et al. [10] suggest a search space for the penalty parameter. Also, the modeling procedure and parameter selection are examined. The findings demonstrate that LS-SVM outperforms neural networks and traditional SVM in terms of prediction accuracy and running speed.

Farmers will be able to gather crop yield information prior to cultivation as a result of the Suresh et al. [11] study, which will aid them in making the right choices. The performance evaluation system for judgment recognition systems using a finite state automation is presented by Otsuki et al. [12]. The amount of word sequences close to the judgment can be used to predict the relationship between the word recognition score and judgment recognition score. A study was undertaken by Kumar et al. [13] to gather the tweets for one month, and around 110 tweets were evaluated based on the expectations, requirements, and needs of the stakeholders. The text mining corpus is created, analyzed, and an algorithm is designed to explain the observed events. The words that appear most frequently and their significance in the events that take place throughout time in the institutional education sector are recognized. The goal of the Yusof et al. [14] study is to forecast how well students in the Bachelor of Usuluddin and Bachelor of Syariah programmers will perform in an English course. When tested against three testing datasets, the proposed deep learning model achieves 93% accuracy in folder classification. We have examined and used Baker and Bang [15], whose paper classified the audio signal into four main emotional states. Using 2000 utterances of the produced audio signal database, we have taken into account various

statistical characteristics of pitch, energy, ZCR (Zero Crossing Rate), and MFCC (Mel frequency cepstral coefficient).

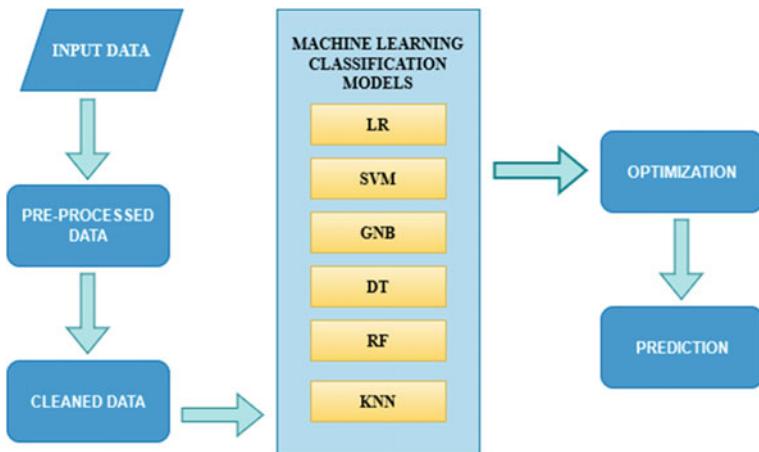
Kusrini et al. [16] The research published in 2019 will cover strategies for paying college costs after the deadline. We noticed two of the many widely used techniques in this field—Naive Bayes and K-Nearest Neighbor (KNN). The official Al-Islam Surakarta Vocational School primary education data for 2017/2018, comprising 236 data, were used for the lab work. In 2020, Barus et al. [17] suggested computing. Churn vaticination studies are conducted using test data amounts ranging from 100, 300, 500, 700, and 900. The findings demonstrate that implementing OpenMP forecasting churn is quicker than routine processing. The information acquired from the dataset taken from the UCI machine learning repository is handled in Tyagi et al. [18]'s publication as they discuss the analysis and classification models utilized in thyroid disease. In order to successfully complete complicated learning activities, such as medical diagnosis and prediction studies, a good knowledge base must be present. This study also suggested several machine learning methods for thyroid prevention diagnostics. Support vector machine (SVM), KNN, decision trees, and machine learning algorithms were employed to forecast the estimated risk of a patient developing thyroid illness. Pushpa et al. [19] used several algorithms to forecast data before settling on the optimal solution. The RandomForest Algorithm is the most effective method for this article based on graphs and metrics. The votes, which are calculated using different algorithms, imply that. Predictive analytics in healthcare are covered in Sarwar et al. [20]'s study, which uses six distinct machine learning methods. For experimental purposes, a dataset of patient medical records is acquired, and six different machine learning algorithms are applied to the dataset. The use of the algorithms is outlined, and their performance and accuracy are compared.

### 3 Proposed Model

In this paper, we suggest a model to forecast the results of students' communication performance levels. We train different machine learning classifiers using the dataset of student grades. The algorithm that proves to be the most accurate, or the one with the best vaticination delicacy, will be compared with the others. The classifier can then be optimized in order to improve the model's prediction accuracy even more. Three levels—2 (Good), 1 (Moderate), and 0—would be available (poor) (Fig. 1).

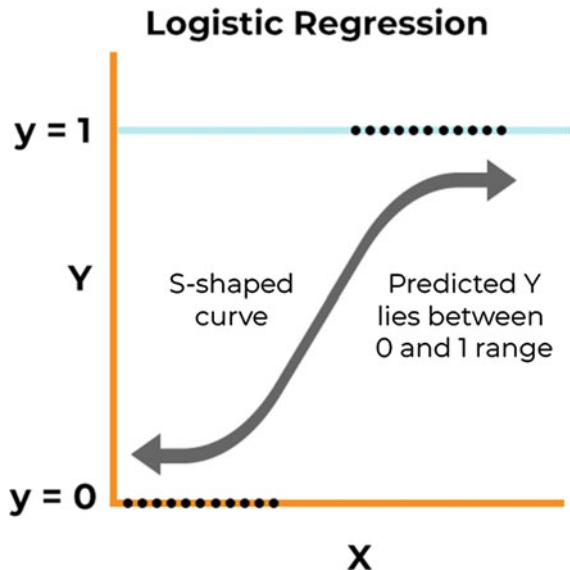
Logistic regression is a supervised learning technique used to forecast discrete or categorical variables. The logistic regression algorithm can be used to solve classification problems in machine learning, and its output can be Yes or No, 0 or 1, Red or Blue, etc. Logistic and linear regression are similar, however, they are used differently. The classification problem is tackled with logistic regression, which predicts discrete values, while the regression problem is solved with linear regression, which predicts continuous values (Fig. 2).

For supervised learning, classification and regression problems can be solved using the K-Nearest Neighbor method. This algorithm works under the assumption



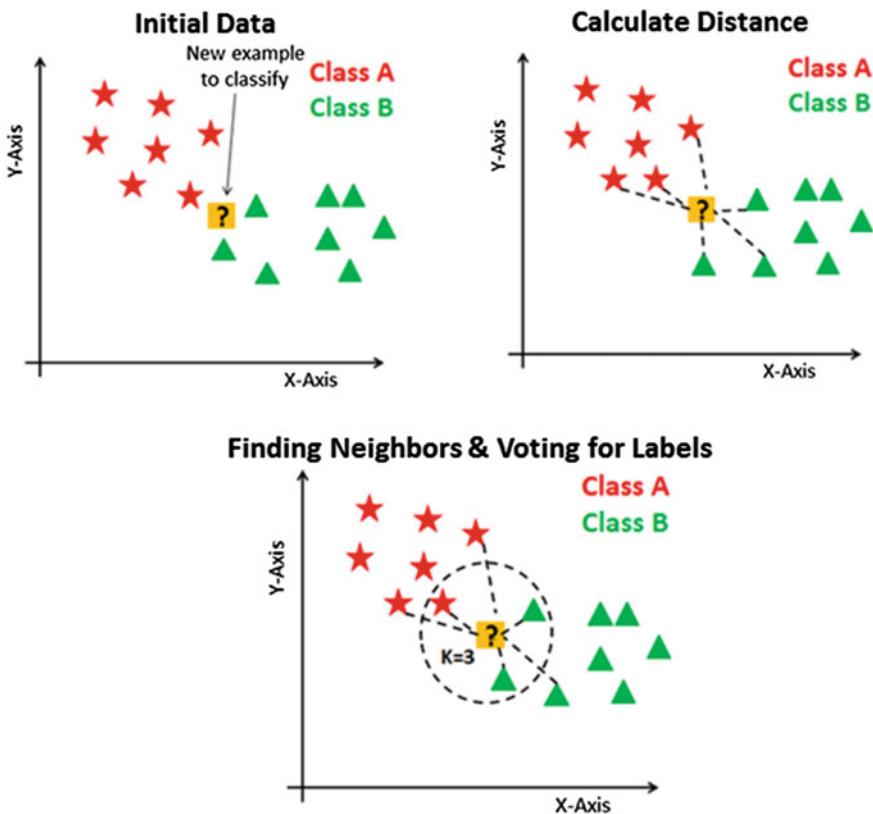
**Fig. 1** Architecture diagram which explains the flow of data that is been trained and predicted

**Fig. 2** Logistic regression



that the new data point and the previous data points are equivalent. The new data points are sorted into the most related categories using these similarities. It is frequently referred to as the lazy learner algorithm since it keeps track of all available datasets and classifies each new instance using K-neighbors (Fig. 3).

Support vector machines, or SVMs, are supervised learning techniques that can be used to solve classification and regression problems. On the other hand, its primary use is categorization problems. SVM tries to build a decision boundary or hyperplane that may divide datasets into discrete groups. The data points that help create the



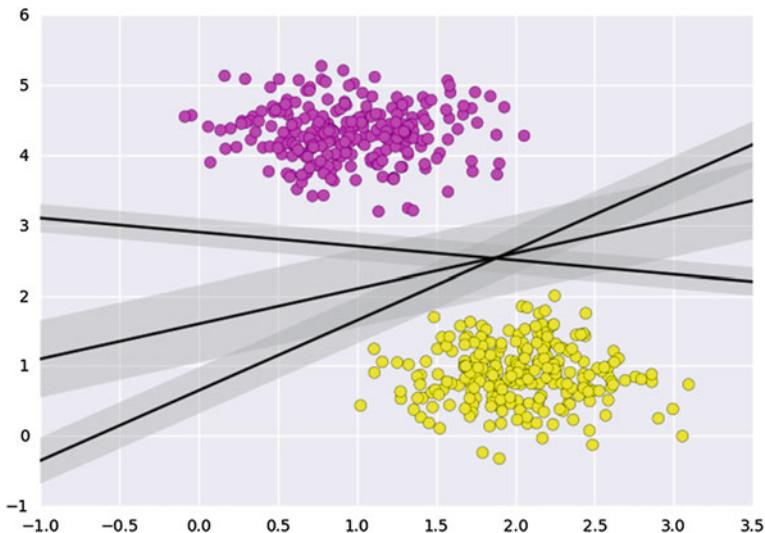
**Fig. 3** K-nearest neighbor

hyperplane and give rise to it, known as support vectors, are what give rise to the name of the support vector machine technique (Fig. 4).

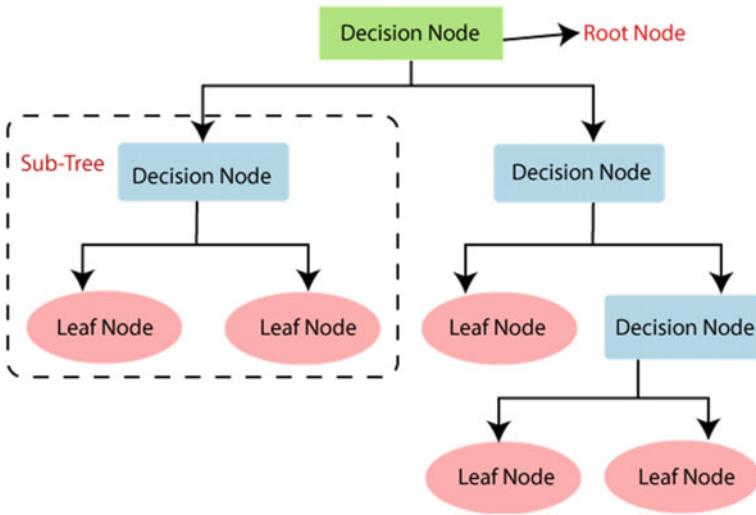
A decision tree is a supervised learning method that is typically used to address classification problems, while it can also be applied to problems with regression. It is compatible with both categorical and continuous variables. It shows a tree-like structure with nodes and branches that grow on additional branches to the leaf node from the base node. The dataset's properties are represented by the core node, while the decision-making processes are shown by the branches, and the leaf nodes show the resolution to the problem (Fig. 5).

Machine learning classification and regression problems may be addressed using the supervised learning method known as random forest. An ensemble learning technique improves the model's performance by making predictions by combining numerous classifiers (Fig. 6).

A supervised learning system called the Naive Bayes classifier makes future predictions using object probability. The Naive Bayes is based on the Bayes theorem



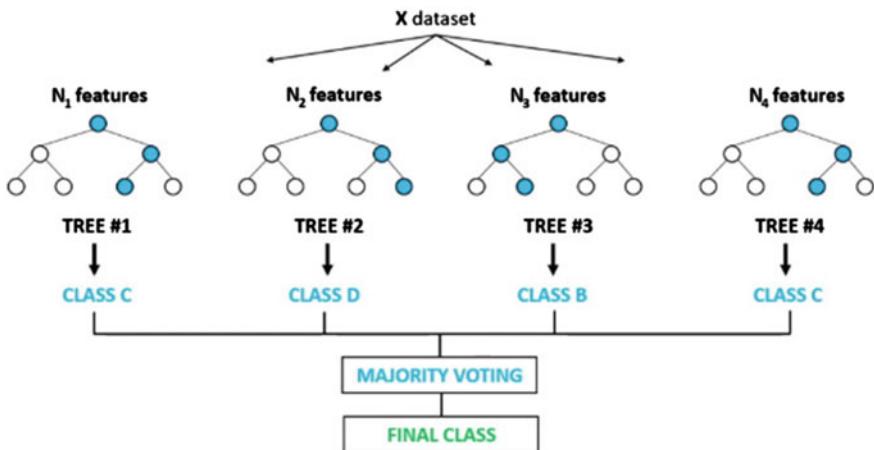
**Fig. 4** Support vector machine



**Fig. 5** Decision tree

and follows the naive assumption, which states that variables are independent of one another (Fig. 7).

The communication testing software provides prediction data. Each student's Scores and Levels are included in the real-time data. About four characteristics are present at Levels 2 (Excellent), 1 (Moderate), and 0. (poor). We obtain about 1–2



**Fig. 6** Random forest classifier

**Fig. 7** Naïve Bayes

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

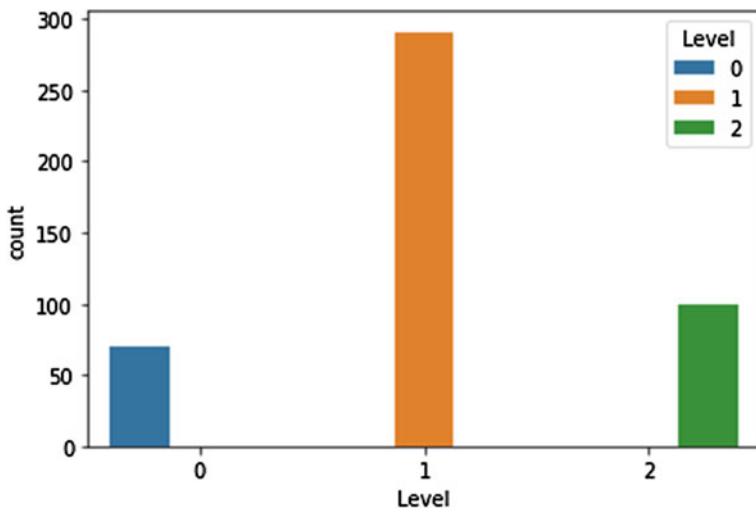
↑                                           ↑  
Likelihood                                  Class Prior Probability  
↓                                           ↓  
Posterior Probability                      Predictor Prior Probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

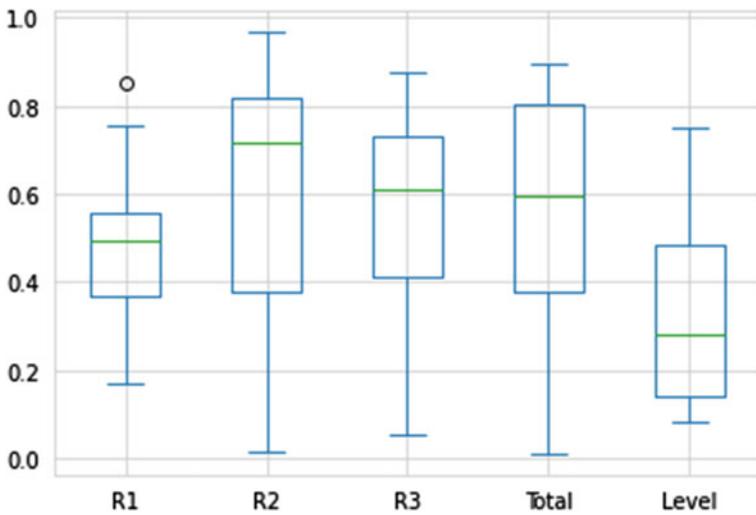
attributes that will forecast the outcomes after filtering these features. The dataset has 460 rows. The observation counts for each category variable are shown using bars in a count plot. The count plot has provided the count for each value, namely 0, 1, and 2, because our Level is the dependent variable or outcome. The observation counts for each category bin are shown using bars in a count plot. The count plot has provided the count for each of the values, namely 2 (Excellent), 1 (Moderate), and 0 because our Level is the dependent variable or the outcome (poor). Some attributes are less important or irrelevant for predicting the result. So, data cleaning is carried out to retain only those relevant attributes for prediction (Figs. 8 and 9).

It helps us to understand the way in which data has been spread out. We have split the data into training and testing to fit the model and get the desired outcome. They've been resolved on the probabilities 80 and 20 (Fig. 10).

The rectangles in this histogram, where each bar represents some data, are shown. It is used in numerous branches of mathematics, including statistics. In statistical data, frequency is nothing more than the repeating of numbers. Moreover, it can be



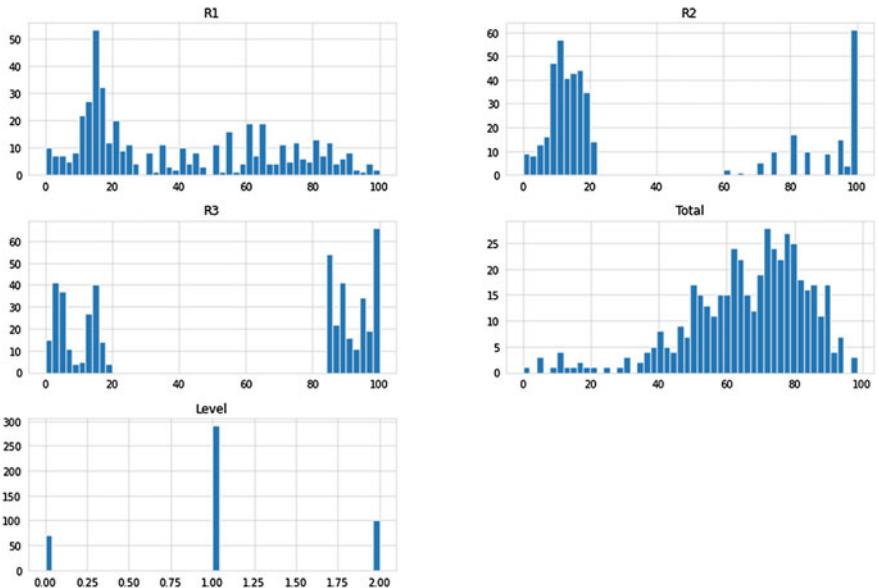
**Fig. 8** Count plot for the dataset provided, it shows the three categories that have been divided into 2 (excellent), 1 (moderate), and 0 (poor)



**Fig. 9** Shows a boxplot for the point-gauged attributes. We use box plots to display the distribution of data in a standardized form

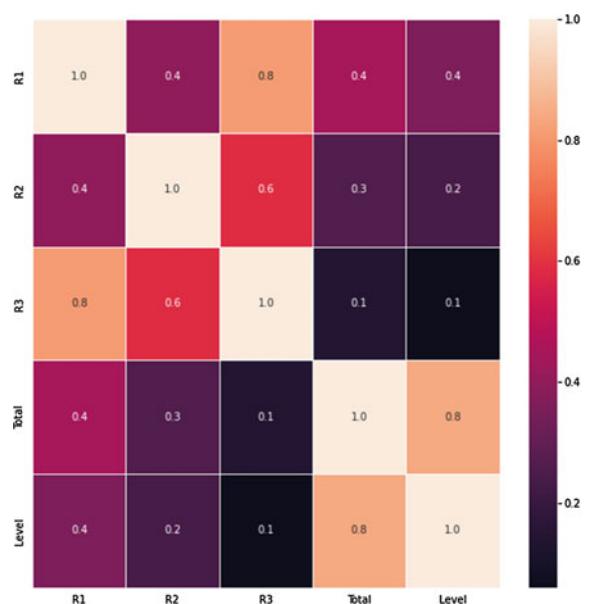
represented as a frequency distribution table. Several different styles of graphs can be used to depict the frequency distribution. The histogram is the one in the graph (Fig. 11).

One of the most popular methods for displaying the dataset is a heatmap. It aids in our comprehension of the characteristic that is most closely associated. It utilizes



**Fig. 10** The distribution of the data is shown in the histogram

**Fig. 11** Visualization using heatmap



a technique of color coding to include numerous values. On the heatmap of the functionality attributes in our dataset, the correlation between the attributes is displayed. By using color coding, we can see which traits are most closely related to different Levels.

## 4 Results and Discussion

Model training, pattern recognition, testing, and evaluation outcomes make up this process. The data set was split into testing and training sets, as was previously described. The model is built using the bracket techniques in the training set. Results will then be assessed. We tried the following six classification methods to see which one will work best for prediction: Logistic Regression (LR), Support Vector Machines (SVM), Gaussian Naive Bayes (GNB), Decision Tree (DT), Random Forest (RF), and K-Nearest Neighbors (KNN). Five common metrics—Accuracy, Recall, Precision, Kappa, and F-measure—are employed in the experiment to rate the effectiveness of the categorization model (Table 1).

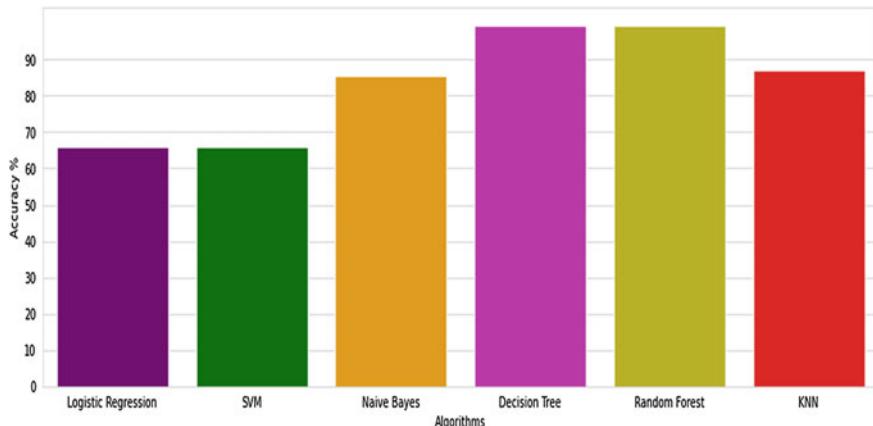
To get the highest level of precision, the experiment is run. In this essay, we make use of information from an application's Student's Communication test. It is carried out to see if the quantity of training data affects prediction precision. When combined with training data, the decision tree and random forest algorithms predict the student's communication level with the best degree of accuracy. Hence, normalization took place during the pre-processing data stage (Fig. 12).

## 5 Conclusion and Future Work

This research compares various categorization techniques for predicting communication effectiveness. Several bracket models are available, each of which is based on a particular feeling and methodology. The accuracy of the six most popular classifiers—Logistic Regression (LR), Support Vector Machines (SVM), Gaussian Naive

**Table 1** Accuracy comparison between different algorithms

Algorithm	Accuracy
LR	0.66
SVM	0.66
GNB	0.85
DT	0.99
RF	0.99
KNN	0.86



**Fig. 12** Accuracy comparison between different algorithms

Bayes (GNB), Decision Tree (DT), Random Forest (RF), and K-Nearest Neighbors—was compared using many datasets (KNN). Depending on the dimensions and properties of the data sets, each model performs differently. The algorithms' performance measurements only slightly differed from one another. To evaluate which method was the most effective overall, a table was made with each performance metric against each dataset and each algorithm. We drew a graphical representation of the scores in chance form to help us comprehend them better. We concluded from our analysis of the data that the Decision Tree and Random Forest algorithms outperform all others in terms of accuracy. KNN, which is followed by Naive Bayes, Logistic Regression, and SVM, is the second-best classifier. The main idea of this work was to select the fashionable classifier from the most common approaches. But, in the future study, different models might be taken into comparison and selection. In addition to the one mentioned other interesting noise reduction techniques could be used to improve the outcomes of this study. The use of colorful measurements to compare the algorithms' performance is another component that would be intriguing to include in any further investigation.

## References

1. Santhana Krishnan J, Geetha S (2019) Prediction of heart disease using machine learning algorithms. 2019 1st international conference on innovations in information and communication technology (ICIICT), pp 1–5. <https://doi.org/10.1109/ICIICT1.2019.8741465>
2. Varma A, Sarma A, Doshi S, Nair R (2018) House price prediction using machine learning and neural networks. 2018 Second international conference on inventive communication and computational technologies (ICICCT), pp 1936–1939. <https://doi.org/10.1109/ICICCT.2018.8473231>

3. Joy LC, Raj A (2019) A review on student placement chance prediction. 2019 5th International conference on advanced computing & communication systems (ICACCS), pp 542–545. <https://doi.org/10.1109/ICACCS.2019.8728505>
4. Ying E (2012) Application of support vector regression algorithm in colleges recruiting students prediction. 2012 International conference on computer science and electronics engineering, pp 173–176. <https://doi.org/10.1109/ICCSEE.2012.456>
5. Bujang SDA et al (2021) Multiclass prediction model for student grade prediction using machine learning. IEEE Access 9:95608–95621. <https://doi.org/10.1109/ACCESS.2021.3093563>
6. Wei X (2011) Student achievement prediction based on artificial neural network. 2011 International conference on internet computing and information services, pp 485–487. <https://doi.org/10.1109/ICICIS.2011.126>
7. Lee MF, Nawi NFM, Lai CS (2017) Engineering students' job performance prediction model based on adversity quotient & career interest. 2017 7th World engineering education forum (WEEF), pp 132–135. <https://doi.org/10.1109/WEEF.2017.8467175>
8. Gull H, Saqib M, Iqbal SZ, Saeed S (2020) Improving learning experience of students by early prediction of student performance using machine learning. 2020 IEEE international conference for innovation in technology (INOCON), pp 1–4. <https://doi.org/10.1109/INOCON50539.2020.9298266>
9. Ju S, Zhang Z, Zhu X (2020) Video prediction strategy based on Markov modified model. 2020 IEEE 3rd international conference on information systems and computer aided education (ICISCAE), pp 105–109. <https://doi.org/10.1109/ICISCAE51034.2020.9236861>
10. Qiu G, Liao L, Wu Z, Du Q (2011) Thunderstorm prediction study based on PCA and least square support vector machine. 2011 International conference on consumer electronics, communications and networks (CECNet), 2011, pp 2828–2831. <https://doi.org/10.1109/CECNET.2011.5768933>
11. Suresh N et al. (2021) Crop yield prediction using random forest algorithm. 2021 7th International conference on advanced computing and communication systems (ICACCS), pp 279–282. <https://doi.org/10.1109/ICACCS51430.2021.9441871>
12. Otsuki T, Ito A, Makino S, Otomo T (1994) The performance prediction method on sentence recognition system using a finite state automaton. Proceedings of ICASSP '94. IEEE international conference on acoustics, speech and signal processing, pp I/397-I/400 vol 1. <https://doi.org/10.1109/ICASSP.1994.389272>
13. Kumar KS, Geetha DE, Kumar TVS (2014) Prediction of events in education institutions using online social networks. International conference on circuits, communication, control and computing, pp 412–415. <https://doi.org/10.1109/CIMCA.2014.7057834>
14. Yusof MHM, Khalid IA (2021) Precision education reviews: a case study on predicting student's performance using feed forward neural network. 2021 international conference of technology, science and administration (ICTSA), pp 1–4. <https://doi.org/10.1109/ICTSA52017.2021.9406525>
15. Bhakre SK, Bang A (2016) Emotion recognition on the basis of audio signal using Naïve Bayes classifier. 2016 International conference on advances in computing, communications and informatics (ICACCI), pp 2363–2367. <https://doi.org/10.1109/ICACCI.2016.7732408>
16. Kusrini K, Luthfi ET, Muqorobin M, Abdullah RW (2019) Comparison of Naïve Bayes and K-NN method on tuition fee payment overdue prediction. 2019 4th International conference on information technology, information systems and electrical engineering (ICITISEE), pp 125–130. <https://doi.org/10.1109/ICITISEE48480.2019.9003782>
17. Barus DT, Elfarizy R, Masri F, Gunawan PH (2020) Parallel programming of churn prediction using Gaussian Naïve Bayes. 2020 8th International conference on information and communication technology (ICoICT), pp 1–4. <https://doi.org/10.1109/ICoICT49345.2020.9166319>
18. Tyagi A, Mehra R, Saxena A (2018) Interactive thyroid disease prediction system using machine learning technique. 2018 Fifth International conference on parallel, distributed and grid computing (PDGC), pp 689–693. <https://doi.org/10.1109/PDGC.2018.8745910>

19. Pushpa K, Manjunath TN, Mrunal TV, Singh A, Suhas C (2017) Class result prediction using machine learning. 2017 International conference on smart technologies for smart nation (SmartTechCon), pp 1208–1212.<https://doi.org/10.1109/SmartTechCon.2017.8358559>
20. Sarwar MA, Kamal N, Hamid W, Shah MA (2018) Prediction of diabetes using machine learning algorithms in healthcare. 2018 24th International conference on automation and computing (ICAC), pp 1–6. <https://doi.org/10.23919/IConAC.2018.8748992>

# A Dynamic Threshold-Based Cognitive Radio Energy Detection Algorithm for Multi-carrier Waveforms



Jyoti Gupta, Ankit Gupta, Sachin Chauhan, and Rahul Srivastava

**Abstract** Proper spectrum utilisation is a crucial problem in radio applications. It is estimated that around seventy percent of the spectrum is wasted, which needs to be accessed to effectively perform cellular systems. Cognitive radio (CR) is an advanced technique for obtaining the spectrum efficiently. Spectrum Sensing (SS) is the most significant component of CR. In this work, we have designed a novel Energy Detection (ED) algorithm for an advanced waveform. The presented ED is compared with the conventional ED and Match Filter (MF). The results show that the proposed ED outclasses the conventional ED and MF algorithms and obtains significant throughput with low computational complexity.

**Keywords** Cognitive radio · Energy detection · Spectrum sensing · False alarm

## 1 Introduction

The utilisation of radio applications is increasing daily, so congestion in the spectrum is seen as a severe concern [1]. It is seen that the scarcity of the spectrum is one of the main causes due to which the Quality of Service (QoS) in radio applications may degrade and may not perform well. Currently, the static spectrum policy of the Federal Communication Commission (FCC) does not permit sharing of the bandwidth dynamically with other users and applications [2]. CR is an advanced technology that may upsurge the spectrum capability by dynamically sharing the spectrum between users [3]. The Primary Users (PU) and Secondary Users (SU) are licensed and non-licensed users. CR usually senses the idle spectrum in PU and allocates it to the SU based on the availability of the PU [4]. SS is an important method for CR, which helps find the unused spectrum. SS identifies and utilises the idle bandwidth better by allocating it to SU [5]. ED is a simple detector that identifies the user by estimating the threshold values of the received energy signal and a pre-calculated threshold value [6]. It is noted that the ED can detect the idle range

---

J. Gupta (✉) · A. Gupta · S. Chauhan · R. Srivastava  
Arya College of Engineering and I.T., Kukas, India  
e-mail: [pjyotigupta1712@gmail.com](mailto:pjyotigupta1712@gmail.com)

of the spectrum with a high Signal-to-Noise Ratio (SNR), which is considered one of the drawbacks of the conventional ED [7]. This work presents an advanced ED based on dynamic threshold detection. The proposed algorithm obtained a detection at low SNR. In [8], the authors designed a SS method by estimating the threshold of the received signal. It is observed that the presented method acquired a gain of 1dB as associated with the standing procedures. However, the system's intricacy is not discussed, but it is observed that the implementation of ED is not as complex as compared with other algorithms. The proposed article introduced a novel method to overcome the effect of false alarms in signal detection [9]. The study's results revealed that the selection of the threshold value based on the distortion level of the signal has the highest probability of enhancing the detection efficiency of the framework. The complexity upsurges with the surge in the number of detected trials. In [10], the authors introduced a dual-stage match-filter algorithm for the 5G waveforms. It is seen that the presented algorithm effectively enhances the performance of the system even in a fading environment. The intricacy of the different SS algorithms is analysed by estimating the number of arithmetic needed for finding the required signals. It is renowned that the anticipated dual-stage Matched Filter (MF) complication is greater than that of conventional algorithms. The presented article [5] introduced a novel Cr-cantered SS technique to enhance the spectrum efficiency of the framework. The experimental graphs divulge that the projected algorithm obtained significant performance improvements by reducing the interference between PU and SU. It is also seen that the prior estimation of the channel information of the system improves the detection performance with high fidelity. In [11], the authors have analysed the impact of noise on the SS algorithms. A novel ED is designed with several antennas, which reduce the effect of the noise in the sign and improve the recognition of the gain of the outline. Self-integration into the multiplicity of the wireless environment is a CR capacity or shortened as 5G integrates and connects all wireless technologies through WISDOM, and CR adapts to and utilises all wireless technologies. The fifth generation of cellular wireless standards, 5G, and cognitive radio, CR, are potential future technologies [12]. On the one hand [13], CR allows users to improve their spectrum efficiency using free licensed spectrum holes. The current method of accessing the spectrum, also known as an approach, is based on fixed spectrum allocation. Because the available spectrum has already been allotted to various uses, deploying increasingly large and diversified wireless applications has resulted in a spectrum shortage [14]. Although there is a spectrum shortage, research has previously shown that more than 75% of it is still unused. As a result [15], the usage of the spectrum has inspired many researchers and given rise to the idea of dynamic spectrum access, which allows a secondary user to access a primary user's allotted spectrum without interfering with it. The algorithm [16] increases the accuracy of energy detection by using the number of subcarriers that are used and the noise level in multi-carrier waveforms. In [17] the result, the interference of primary users can be reduced, and the available spectrum can be used more effectively. The author says [18] that the noise power is analysed using the power of received signal on unused subcarriers. A dynamic threshold is determined by the probability of a false alarm (PFA) and depends on the estimated noise power. The PFA is calculated as noise

power depending on the required detection performance [19, 20]. If the energy on any subcarriers surpasses the dynamic threshold, the presence of the primary user is detected [21].

## 2 System Model

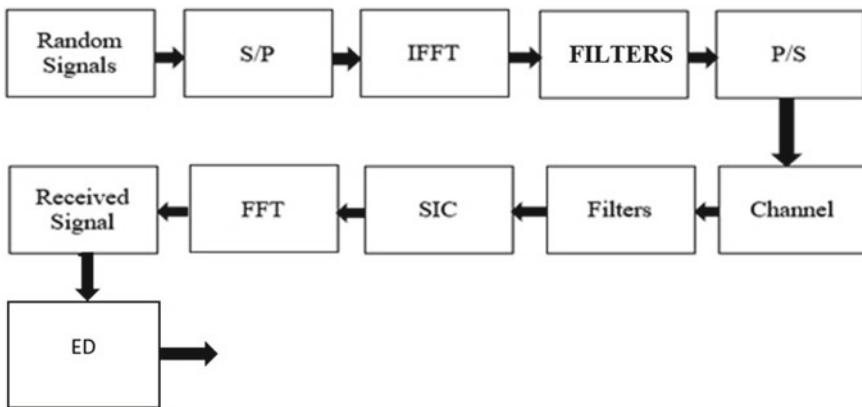
This section introduces a novel ED algorithm for the Multi-Carrier Waveform (MCW), commonly known as Filter Bank Multi-Carrier (FBMC). FBMC is based on the MCW and is considered one of the most advanced candidates for the advanced radio framework. The FBMC consists of filters that reduce interference between subcarriers and enhance spectrum efficiency [12]. The proposed ED is applied at the receiving terminal of the FBMC, which estimates the predetermined and received signal thresholds for the estimation of the idle spectrum. The schematic of FBMC and ED is given in Figs. 1 and 2.

The availability and non-availability of the idle spectrum are denoted  $H_0$  &  $H_1$  with noise variance  $N_n$  [14].

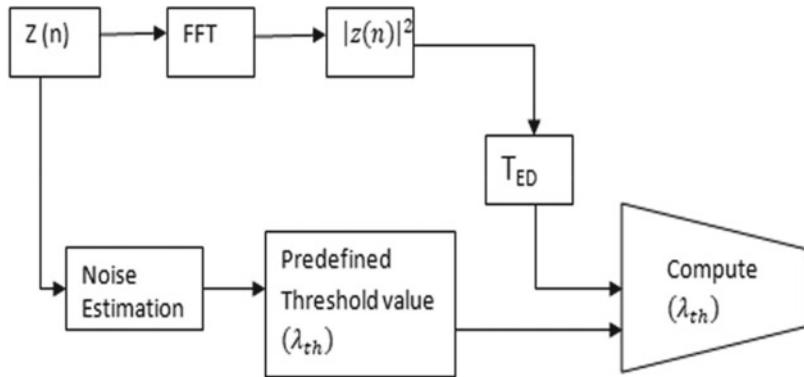
$$H_0 : y(n) = N_n \quad (1)$$

$$H_1 : y(n) = z(n) + N_n \quad (2)$$

Threshold estimation is one of the important steps in the SS algorithm. It is given by:



**Fig. 1** Schematic of FBMC



**Fig. 2** Schematic of ED

$$E_{TH} = \sum_{n=1}^N (y(n))^2 \quad (3)$$

The signal is computed through the pre-estimated threshold ( $\lambda_{th}$ ) and the idle spectrum is determined as given below:

$$T_{ED} < \lambda_{th} : \text{Non Availability of } Pu \quad (4)$$

$$T_{ED} \geq \lambda_{th} : \text{Availability of } Pu \quad (5)$$

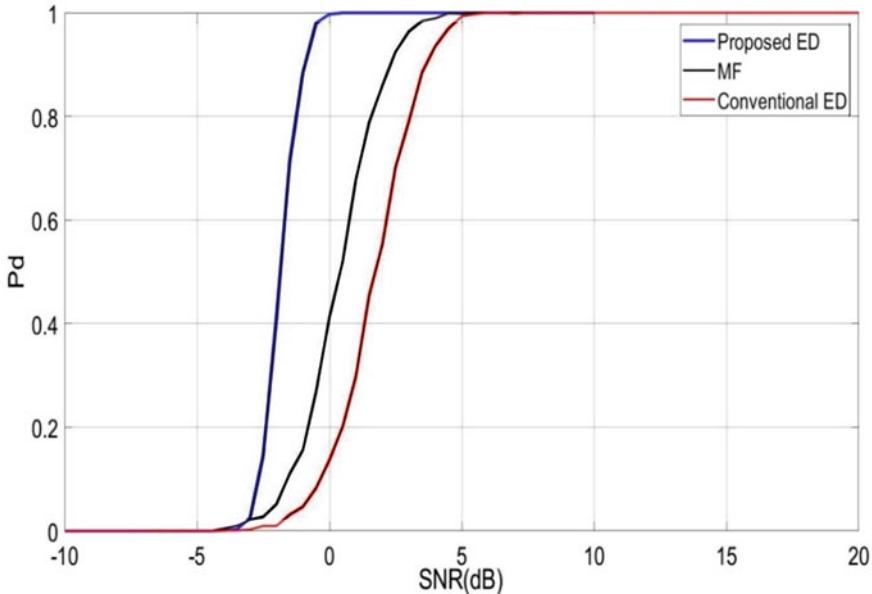
### 3 Simulation Results

We have analysed and estimated the performance of the advanced ED algorithm by using a computer simulation in MATLAB. Table 1 shows the simulation parameters of the presented ED algorithm.

In Fig. 3, we investigated the detection effectiveness of the projected ED with the conventional ED and MF algorithms. The graph is between SNR and Probability of

**Table 1** Imitation constraints

S. No	Constraints
1	FBMC waveform
2	ED algorithm
3	Samples N = 200
4	Rayleigh channel
5	64 subcarriers



**Fig. 3** SNR Vs Pd

detection ( $P_d$ ). The free spectrum estimation is obtained at a low SNR of  $-2$  dB compared with the conventional ED ( $5$  dB) and MF ( $2.3$  dB). However, it was noted that the increase in samples can also increase the complexity of the framework.

The false alarm estimation of SS methods is studied, analysed, and presented in Fig. 4. The graph is between  $P_{fa}$  and  $P_d$ . It is observed that the proposed ED estimates the noise variance and performs an accurate detection of the signal.

The effectiveness analysis of the anticipated framework is estimated in Fig. 5. The graph shows the relation between SNR (dB) and BER (bit error rate). The BER of  $10^{-3}$  is acquired at a low SNR of  $7$  dB as compared with the conventional ED (SNR =  $10$  dB) and MF methods (SNR =  $8.2$  dB). In Fig. 5, the blue line dot shows the proposed ED, Dark red line shows MF, and the red line dot shows conventional ED.

The Power Spectrum Density (PSD) is given in Fig. 6. It is found that the spectrum leakage for  $k = 1, 2, 3$ , and  $4$  is given by  $-250, -750, -1600$ , and  $-2300$ . Hence, it is seen that spectral efficiency is obtained for the given FBMC. The FBMC related to Filter Bank Multicarrier highlights the merits of the modulation scheme of the user in Fifth generation (5G) communication system.

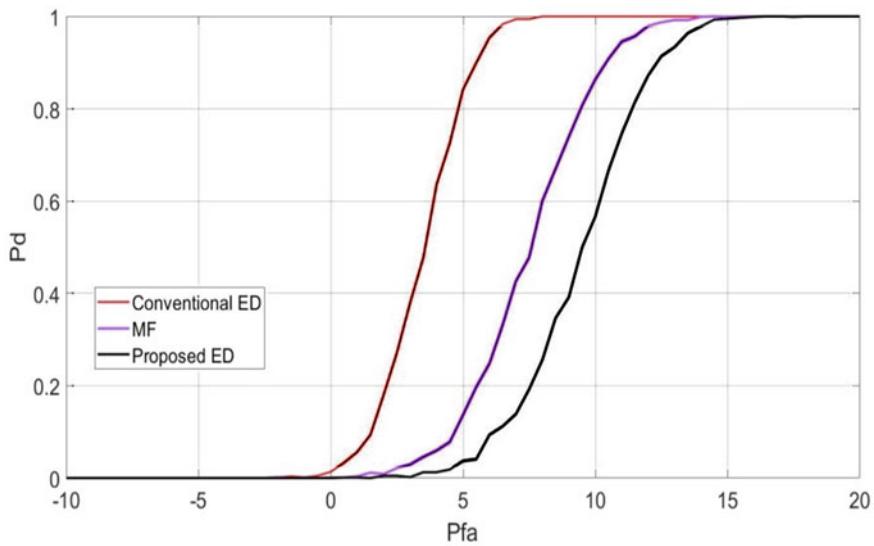


Fig. 4 Pfa Vs Pd

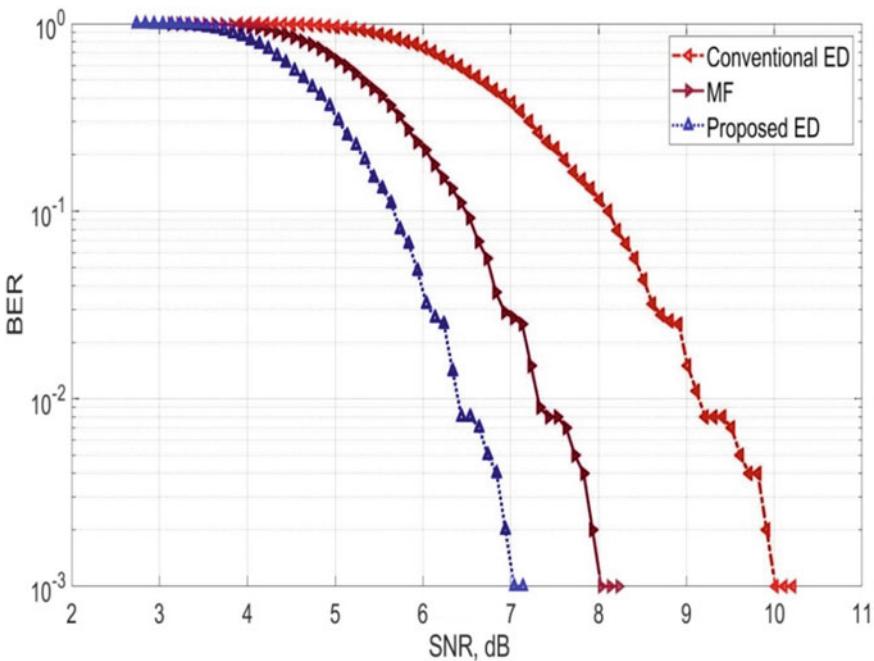
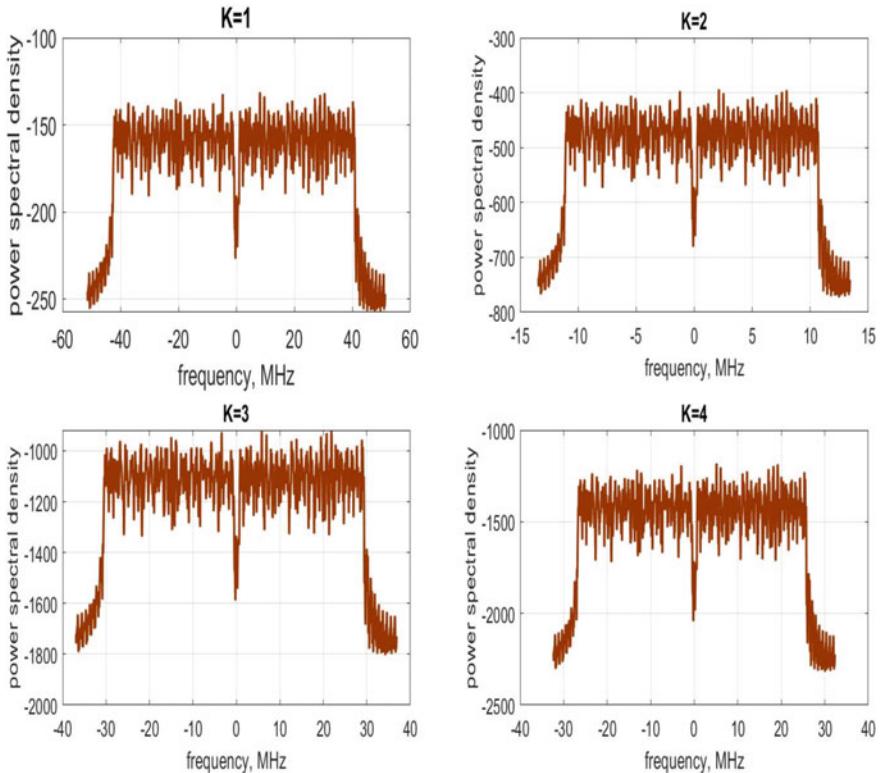


Fig. 5 BER



**Fig. 6** PSD of FBMC

## 4 Conclusion

The article proposed a novel CR-based ED algorithm for the FBMC waveform. The ED enhances the spectrum performance of the framework by detecting the unused spectrum from the PU and allocating it to the SU without creating any disturbance between the PU and SU. It is noted that the detection of the idle spectrum based on the dynamic threshold detection can improve the spectrum efficacy of the scheme. Hence, the noise modification estimation advances the CR system's gain. In conclusion, as results show that the proposed ED algorithm which is used for multi-carrier waveforms is the most efficient and steady technique to detect PUs in OFDM signals.

## References

1. Kumar A, Gupta M (2018) A review on activities of fifth generation mobile communication system. *Alex Eng J* 57(2):1125–1135
2. Chen S, Law R, Zhang M, Si Y (1804) Mobile communications for tourism and hospitality: a review of historical evolution, present status, and future trends. *Electronics* 2021:10
3. Kumar A, Sharma H (2022) Intelligent cognitive radio spectrum sensing based on energy detection for advanced waveforms. *Radioelectron Commun Syst* 65:149–154
4. Kumar A, Nandha Kumar P (2019) OFDM system with cyclostationary feature detection spectrum sensing. *ICT Express* 5(1):21–25
5. Pandya P, Durvеш A, Parekh N (2015) Energy detection based spectrum sensing for cognitive radio network. 2015 Fifth international conference on communication systems and network technologies.<https://doi.org/10.1109/CSNT.2015.264>
6. Dao N-N, Na W, Tran A-T, Nguyen DN, Cho S (2020) Energy-efficient spectrum sensing for IoT devices. *IEEE Syst J* 15:1077–1085
7. Nandhakumar P, Kumar A (2016) Analysis of OFDM system with energy detection spectrum sensing. *Indian J Sci Technol* 9:1–6
8. Kumar A, Sharma MK, Sengar K, Kumar S (2020) NOMA based CR for QAM-64 and QAM-256. *Egypt Inf J* 21(2):67–71
9. Martínez Plata DM, Andrade Reátega ÁG (2012) Evaluation of energy detection for spectrum sensing based on the dynamic selection of detection-threshold. *Procedia Eng* 35:135–143
10. Kumar A, Venkatesh J, Gaur N, Alsharif MH, Jahid A, Raju K (2023) Analysis of hybrid spectrum sensing for 5G and 6G waveforms. *Electronics* 12(1):138
11. Luo J, Zhang G, Yan C (2022) An energy detection-based spectrum-sensing method for cognitive radio, Article ID 3933336. <https://doi.org/10.1155/2022/3933336>
12. Xuping Z, Jianguo P (2007) Energy-detection based spectrum sensing for cognitive radio. 2007 IET conference on wireless, mobile and sensor networks (CCWMSN07), pp 944–947. <https://doi.org/10.1049/cp:20070306>
13. Zeng Y, Liang YC, Hoang AT et al (2010) A review on spectrum sensing for cognitive radio: challenges and solutions. *EURASIP J Adv Signal Process* 2010:381465. <https://doi.org/10.1155/2010/381465>
14. Zhang W, Ranjan KM (2009) Optimization of cooperative spectrum sensor using energy detection in cognitive radio networks. *IEEE Trans Wirel Commun* 8(12):5761–5766
15. Shi Z, Yin Z, Cheng Q (2015) Novel spectrum sensor using algorithms for OFDM cognitive radio networks. *Sensors* 15:13966–13993
16. Yu G, Long H, Yue G (2009) Dynamic threshold based spectrum detection in cognitive radio systems. In: Proceedings of the 2009 5th international conference on wireless communications, networking and mobile computing, Beijing, China, 24–26 September 2009, pp 1–4
17. Tellambura C, Jiang H (2011) Energy detection based cooperative spectrum sensing in cognitive radio networks. *IEEE Trans Wirel Commun* 10(4):1232–1241
18. Oborina A, Moisio M, Koivunen V (2012) Visa, efficiency, performance and effectiveness of Mobile MIMO OFDM systems with application to UTRAN LTE downlink. *IEEE Trans Wirel Trans* 11(8):2696–2706
19. van Nee R, Prasad R (2000) OFDM for mobile multimedia transmissions. Artech House
20. Ju H, Cho E, Kim S-H (2021) Energy-detection based false alarm reduction in polar-coded uplink control channel transmission in 5G-NR. In: Proceedings of the 2021 IEEE 93rd vehicular technology conference (VTC2021-Spring), pp 1–6, Helsinki, Finland
21. Kozal ASB, Merabti M, Bouhafs F (2012) An improved energy detection scheme for cognitive radio networks in low SNR region. In: 2012 IEEE symposium on computers and communications (ISCC), Cappadocia, 1–4 July 2012, pp 684–689.<https://doi.org/10.1109/ISCC.2012.6249377>

# Comparison of Different Alphabetically Shaped Microstrip Patch Antennas in the S-Band Using HFSS



Shubhi Gupta, Sachin Chauhan, Rahul Srivastava, and Ankit Gupta

**Abstract** Antennas play a very important role in our daily lives. Everything that makes our lives easier is connected to antennas, such as telephones, mobiles, television, and security gadgets. Even these antennas are important in the military also. So, we need advancement in currently available sources that can be helpful to invent something more productive. In this paper, we are comparing three microstrip patch antennas, which we have designed in the shape of “E”, “H” and “U”. This comparison helps us to know about the characteristics of the three as individuals and gives an overview that which amongst them is better. Although much research has been done, this paper compares three antennas, and we know a better option for the best results. The frequency is 2.4 GHz, and all three antennas have been simulated in HFSS 15.0 software.

**Keywords** Microstrip patch antenna · Different-shaped antenna · HFSS 15.0

## 1 Introduction

Antennas are a medium that connects one place to another directly or indirectly in the form of wireless communication. Earlier, huge antennas were used for maximum purposes, but they are too difficult to handle and require a lot of space and maintenance [1–3]. Due to new inventions and technological advancements, the new antennas are of small size [4, 5] and easy to operate [6, 7]. A microstrip antenna is one such antenna that can be widely used nowadays, and it is compact in size and easy to handle. This antenna is constructed on the dielectric substrate [8–10], on which a metallic patch is printed [11]. Feeding of the element is done by the coplanar line or coaxial line and is fed through the bottom of the substrate [12, 13]. These antennas are used in radio frequency identification [14], in the military [15], for surveillance [16], wireless communication [17] and many more. This paper consists of microstrip antennas of different shapes designed and simulated in HFSS 15.0 software. The simulation

---

S. Gupta (✉) · S. Chauhan · R. Srivastava · A. Gupta  
Arya College of Engineering & I.T., Jaipur, India  
e-mail: [guptashubhi234@gmail.com](mailto:guptashubhi234@gmail.com)

frequency used is 2.4 GHz. Although much research had already been done, this research compares different shaped antennas that give us the best shape, which gives better results.

## 2 Proposed Antenna Designs

Different shapes of the same antenna being simulated at the same frequency can yield different outputs. So, in this paper, I have designed three microstrip patch antennas which are in shape “E”, “H”, and “U”. All three antennas are being simulated in HFSS 15.0 software at 2.4 GHz frequency, and a comparison is made at the end to know which performs better.

Any change in the shape, size or configuration can result from a change in the antenna output, even if the same frequency has been given. So, designing different shapes and then simulating and observing their response on the same frequency can show the variation that can help analyze the better shape of the same antenna for preferable results.

### 2.1 E-Shaped Antenna

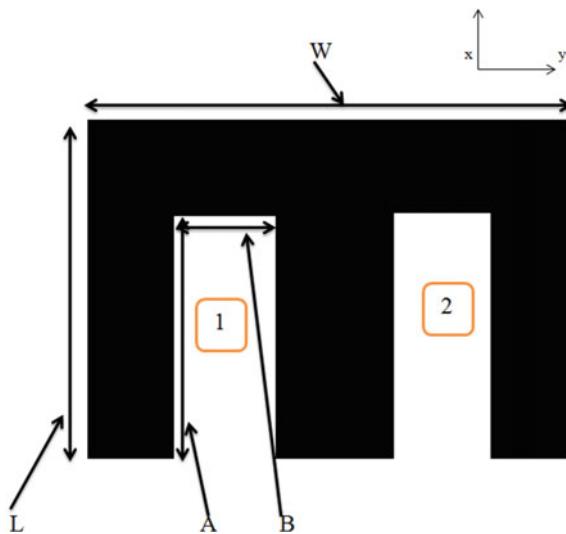
All dimensions are in mm and slot 1 and slot 2 are identical (have exactly same dimensions). A rectangular patch has been designed and then two patches of same dimensions have been cut from the bigger patch to form the shape E (Figs. 1, 2 and Table 1).

### 2.2 H Shaped Antenna

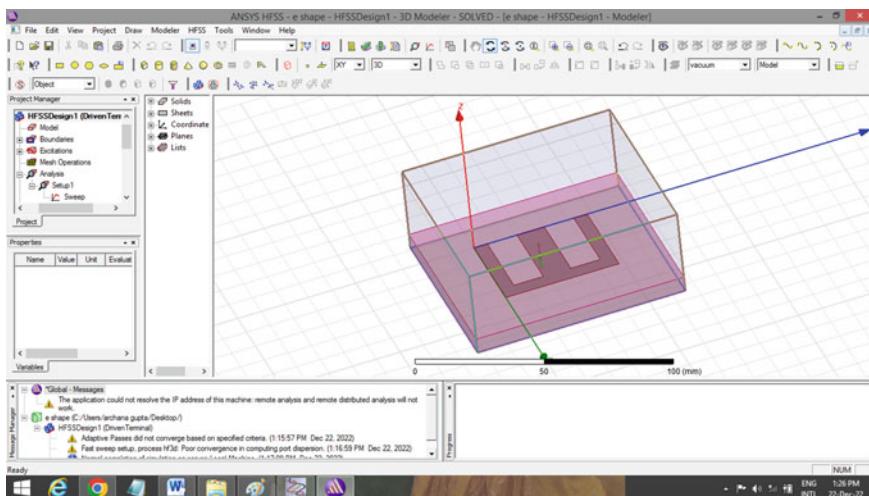
All dimensions are in mm and slot 1 and slot 2 has exactly the same dimensions (are identical). Two identical patches have been cut from the big rectangular patch to form the desired shape H (Figs. 3, 4 and Table 2).

### 2.3 U Shaped Antenna

All dimensions are in mm. Three rectangular patches have been arranged in the shape of U on the substrate to form and simulate the desired shape (Figs. 5, 6 and Table 3).



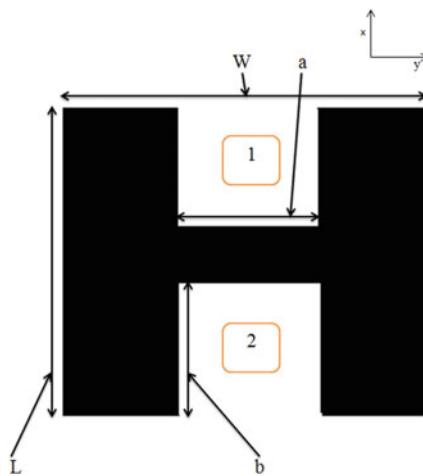
**Fig. 1** E shaped antenna's design



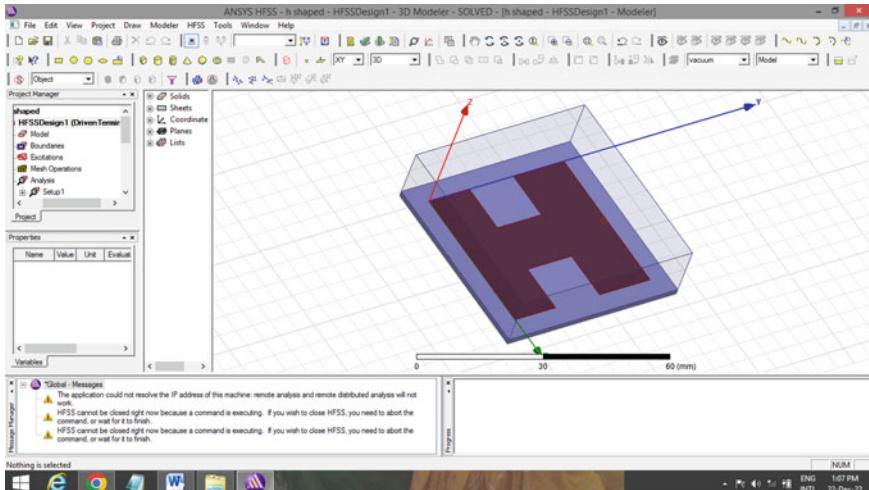
**Fig. 2** Design of E shaped antenna in hfss software

**Table 1** List of dimensions for constructing E shaped antenna

Dimension	W	L	A	B
	36	48	30	12.5



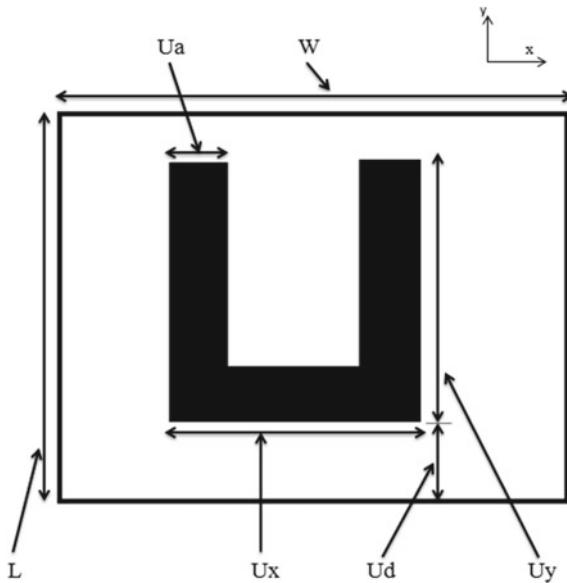
**Fig. 3** H shaped antenna's design



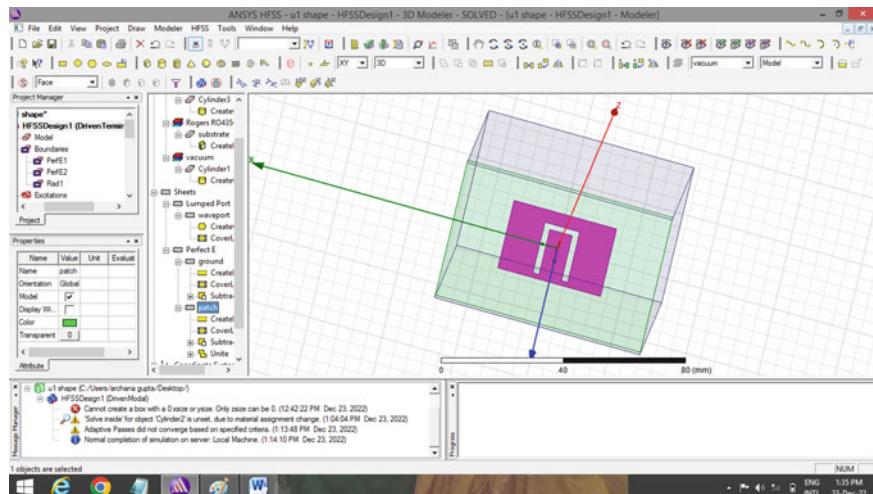
**Fig. 4** Design of H shaped antenna in hfss software

**Table 2** Dimensions for constructing H shaped antenna

Dimension	W	L	a	B
	40	40	13	13



**Fig. 5** U shaped antenna's design



**Fig. 6** Design of U shaped antenna IN hfss software

**Table 3** Dimensions for construction of shape U antenna

Dimension	W	L	Ua	Ud	Ux	Uy
	35.5	26	2.1	4.8	12	19.5

### 3 Simulation Results

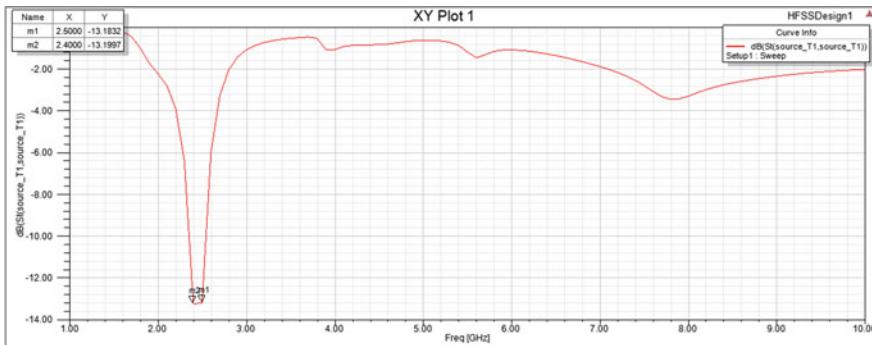
#### 3.1 E Shaped Antenna

Return loss of E shaped micro strip patch antenna is  $-13.19\text{db}$  at a frequency of  $2.4 \text{ GHz}$ . Return loss must be maximum to get less reflection which means we get more power in the output (Figs. 7 and 8).

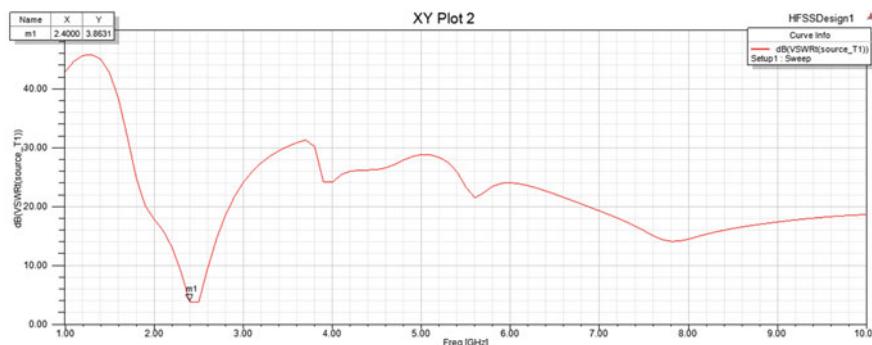
E shaped gives VSWR equals to  $3.8$  at  $2.4 \text{ GHz}$ . VSWR can be expanded as voltage standing wave ratio which gives similar result as that of return loss but on a different scale (Fig. 9).

The gain of E shaped antenna at  $2.4 \text{ GHz}$  frequency is  $5.05\text{db}$ . If an antenna has good gain that means it performs better as the efficiency is better for this antenna and thus the outcome of the receiver antenna is good (Fig. 10).

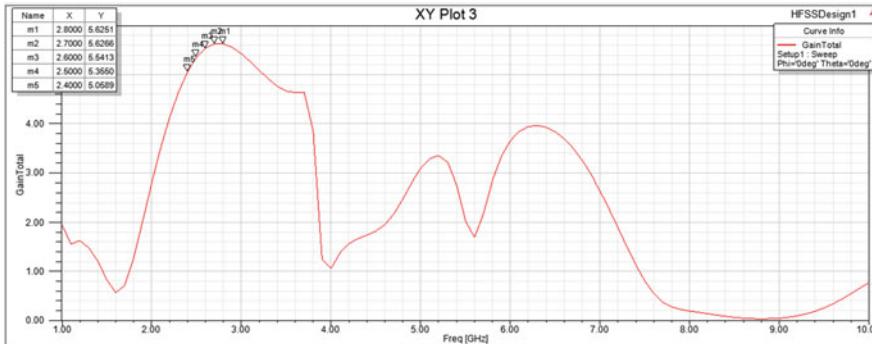
Range of directivity for E shaped antenna is from  $-2.90 \times 10^1$  to  $7.05$ . Directivity helps in the direction in which the radiation of the antenna is mostly concentrated.



**Fig. 7** Return loss graph of E shaped antenna



**Fig. 8** VSWR of E shaped antenna



**Fig. 9** Gain graph of E shaped antenna



**Fig. 10** Directivity of E shaped antenna

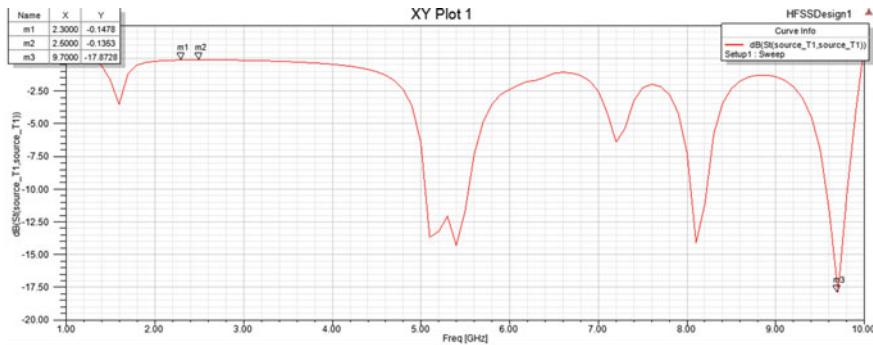
### 3.2 H Shaped Antenna

Return loss of H shaped antenna is -0.14db at frequency of 2.4 GHz and at 9.7 Ghz value of return loss is -17.87db which is less preferable as more the return loss, less the reflection (Figs. 11 and 12).

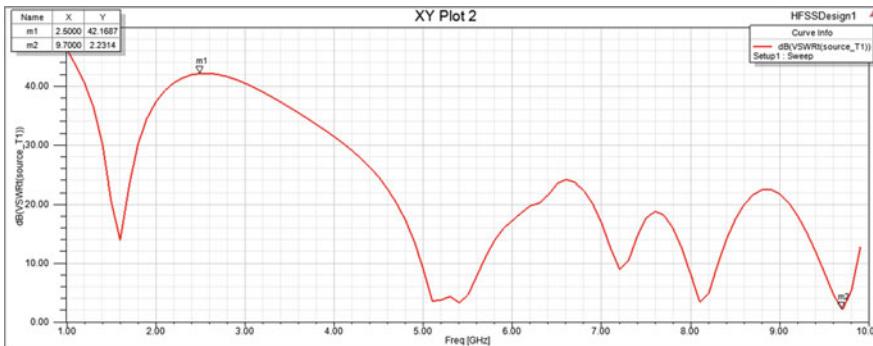
H shaped micro strip patch antenna gives VSWR equals to 42.16 at 2.4 GHz which is not desirable, but it gives 2.2 as VSWR at 9.7 GHz frequency which is better (Fig. 13).

Gain received from H shaped antenna at 2.4 GHz is -10.35db and at 9.7 GHZ frequency gain received is 0.18db which is preferable and better for designing and simulating of antenna (Fig. 14).

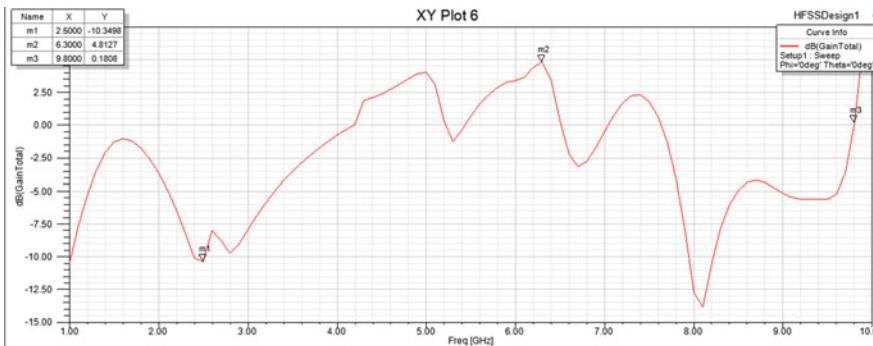
The directivity range for H shaped antenna is from  $-1.82 \times 10^1$  to 4.14. The range of directivity for this antenna is less than previous antenna thus it is less concentrated.



**Fig. 11** Return loss of H shaped antenna



**Fig. 12** VSWR of H shaped antenna



**Fig. 13** Gain of H shaped antenna



**Fig. 14** Directivity of H shaped antenna

### 3.3 U Shaped Antenna

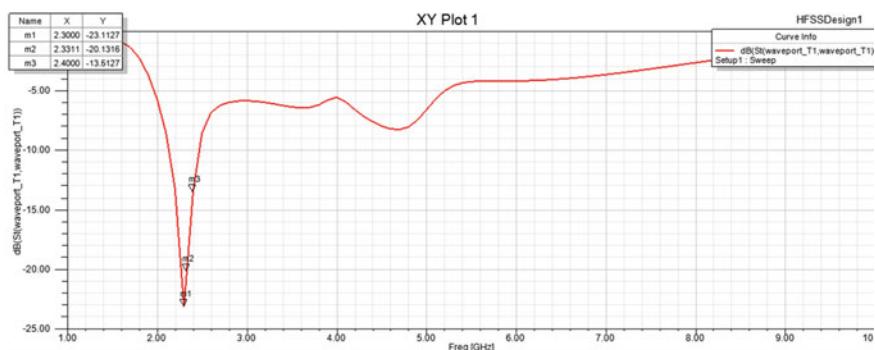
Return loss of U-shaped micro strip patch antenna is  $-13.11\text{db}$  at  $2.4\text{ GHz}$  frequency. Return loss is obtained from all three antennas but this antenna gives better results than the previous two (Figs. 15 and 16).

VSWR of U-shaped antenna is  $3.7$  at  $2.4\text{ GHz}$  frequency and value of VSWR at  $2.3\text{ GHz}$  frequency is  $1.2$  which is preferable (Fig. 17).

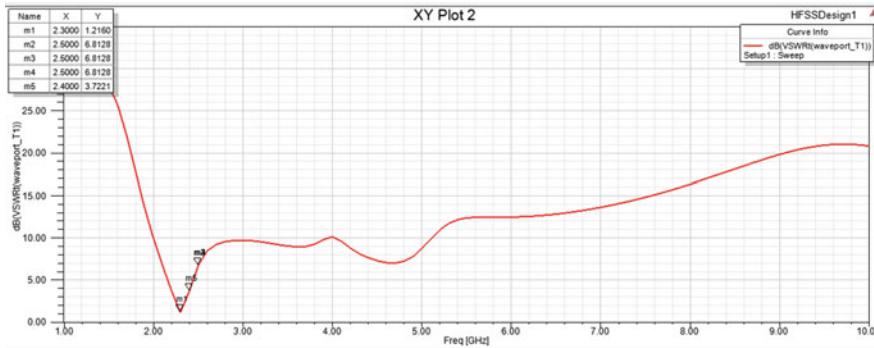
Gain received from U shaped micro strip patch antenna is  $11.3\text{db}$  at  $2.4\text{ GHz}$  frequency and at  $2.3\text{ GHz}$  it is  $9.9\text{ db}$  (Fig. 18).

The range of directivity for U shaped antenna is from  $-1.92 \times 10^1$  to  $3.92$ . This antenna confines in lesser range than the other two thus less concentrated radiations.

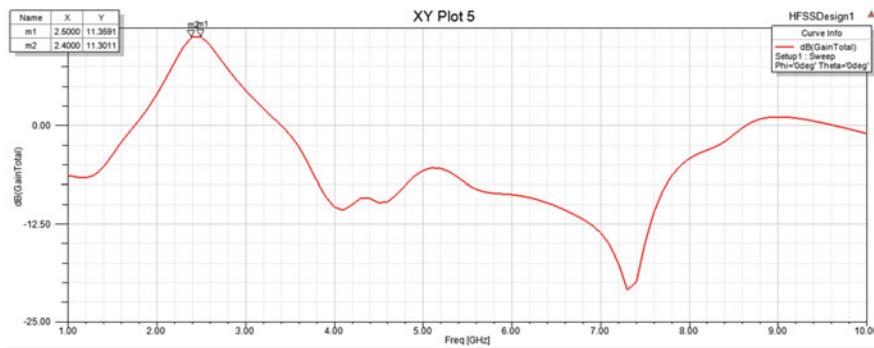
Table 4 shows the behavior of three different antennas at a frequency of  $2.4\text{ GHz}$ .



**Fig. 15** Return loss graph of U shaped antenna



**Fig. 16** VSWR of U shaped antenna



**Fig. 17** Gain of U shaped antenna



**Fig. 18** Directivity of U shaped antenna

**Table 4** Comparison of characteristics of different antennas

Parameter	<b>E shape</b>	<b>H shape</b>	<b>U shape</b>
Return loss	-13.19	-0.14	-13.11
VSWR	3.8	42.16	3.7
Gain	5.05	-10.35	11.3
Directivity	$-2.90 \times 10^1$ to 7.05	$-1.82 \times 10^1$ to -4.14	$-1.92 \times 10^1$ to 3.92

## 4 Conclusion

All the designed antennas are being simulated in HFSS 15.0 software. The performance of antennas is in order as = **E shape > U shape > H shape** at 2.4 GHz frequency. "H" and "U" shaped antennas give better results at frequency of 9.7 GHz and 2.3 GHz respectively. Thus, we can prefer E shaped antennas over the other two for better output and desired results at 2.4 GHz or in S-band.

## References

1. Nguyen DH, Ala-Laurinaho J, Moll J, Krozer V, Zimmer G (2020) Improved sidelobe-suppression microstrip patch antenna array by uniform feeding networks. *IEEE Trans Antennas Propag* 68(11):7339–7347. <https://doi.org/10.1109/TAP.2020.2995416>
2. Hasan R, Ahmed WA, Lu J, Ng HJ, Kissinger D (2019) \$F\\$ -band differential microstrip patch antenna array and waveguide to differential microstrip line transition for FMCW radar sensor. *IEEE Sensors J* 19(15):6486–6496. <https://doi.org/10.1109/JSEN.2019.2909935>
3. Saeidi-Manesh H, Zhang G (2019) Low cross-polarization, high-isolation microstrip patch antenna array for multi-mission applications. *IEEE Access* 7:5026–5033. <https://doi.org/10.1109/ACCESS.2018.2889599>
4. Vishvakarman KS, Mithra K, Kalaiarasan R, Raj KS (2017) Mutual coupling reduction in microstrip patch antenna arrays using parallel coupled-line resonators. *IEEE Antennas Wirel Propag Lett* 16:2146–2149. <https://doi.org/10.1109/LAWP.2017.2700521>
5. Rafique U, Khalil H, Saif-Ur-Rehman (2017) Dual-band microstrip patch antenna array for 5G mobile communications. In: 2017 progress in electromagnetics research symposium - fall (PIERS - FALL), pp 55–59. <https://doi.org/10.1109/PIERS-FALL.2017.8293110>
6. Yang X, Geyi W, Sun H (2017) Optimum design of wireless power transmission system using microstrip patch antenna arrays. *IEEE Antennas Wirel Propag Lett* 16:1824–1827. <https://doi.org/10.1109/LAWP.2017.2682262>
7. Li Y, Luk K-M (2015) 60-GHz substrate integrated waveguide fed cavity-backed aperture-coupled microstrip patch antenna arrays. *IEEE Trans Antennas Propag* 63(3):1075–1085. <https://doi.org/10.1109/TAP.2015.2390228>
8. Arora C, Pattnaik SS, Baral RN (2015) SRR inspired microstrip patch antenna array. *Progr Electromagnetics Res C* 58:89–96. <https://doi.org/10.2528/PIERC15052501>. <http://www.jpier.org/PIERC/pier.php?paper=15052501>
9. Mavaddat A, Armaki SHM, Erfanian AR (2015) Millimeter-wave energy harvesting using \$4times4\$ microstrip patch antenna array. *IEEE Antennas Wirel Propag Lett* 14:515–518. <https://doi.org/10.1109/LAWP.2014.2370103>

10. Casu G, Moraru C, Kovacs A (2014) Design and implementation of microstrip patch antenna array. In: 2014 10th international conference on communications (COMM), pp 1–4. <https://doi.org/10.1109/ICComm.2014.6866738>
11. Sajjad H, Sethi WT, Zeb K, Mairaj A (2014) Microstrip patch antenna array at 3.8 GHz for WiMax and UAV applications. In: 2014 international workshop on antenna technology: small antennas, novel EM structures and materials, and applications (iWAT), pp 107–110. <https://doi.org/10.1109/IWAT.2014.6958609>
12. Phalak K, Sebak A (2014) Aperture coupled microstrip patch antenna array for high gain at millimeter waves. In: 2014 IEEE international conference on communication, networks and satellite (COMNETSAT), pp 13–16. <https://doi.org/10.1109/COMNETSAT.2014.7050517>
13. Koohestani M, Golpour M (2010) U-shaped microstrip patch antenna with novel parasitic tuning stubs for ultra wideband applications. IET Microwaves Antennas Propag 4(7):938–946
14. Islam MT, Shakib MN, Misran N (2009) Broadband EH shaped microstrip patch antenna for wireless systems. Progr Electromagnetics Res 98:163–173
15. Malekpoor H, Jam S (2013) Miniaturised asymmetric E-shaped microstrip patch antenna with folded-patch feed. IET Microwaves Antennas Propag 7(2):85–91
16. Mak CL, Luk KM, Lee KF, Chow YL (2000) Experimental study of a microstrip patch antenna with an L-shaped probe. IEEE Trans Antennas Propag 48(5):777–783
17. Samsuzzaman M, Islam MT, Misran N, Ali MM (2013) Dual band X shape microstrip patch antenna for satellite applications. Procedia Technol 11:1223–1228

# Using MNIST Dataset for De-Pois Attack and Defence



Chetan Niloor , Rashmi Agarwal , and Pradeepa Mishra

**Abstract** Machine learning (ML) has grown into compelling element of diverse systems and applications. Recent research have revealed that ML algorithms are at the risk of serious security vulnerabilities, notwithstanding the outstanding validity of ML algorithms in various signals and decision-making tasks. ML systems trained on user-provided data are vulnerable to various data poisoning attacks, by which attackers inject the malicious training dataset with the purpose of misconducting the learned model. For instance, in model extraction attacks, an attacker can steal the proprietary and secret data of ML models, and in model inversion attacks, they can obtain the private and sensitive data of the training dataset. The objective of this study is to build defence technique by using De-Pois to tackle attackers where they corrupt the ML models by data poisoning in training dataset. De-Pois is the only popular means of protection for poisoning attacks. It offers defence without information of any ML algorithms, kinds of poisoning attacks, and it may be applied to both classification and regression. Hence, De-Pois is a common defence system which will be able to stop all types of attacks. De-Pois approach is crucial for training mimic models, whose goal is to replicate the activities of the target model. The solution developed a De-Pois-based defence mechanism to combat attackers that damage ML models via data poisoning in the training dataset. De-Pois is the only and most popular means of poison defence. De-Pois is a security solution that can be used to safeguard both classification and regression workloads without the need for specialized understanding of ML methods or poisoning attack types.

**Keywords** Machine learning · Data poisoning · Agnostic defense

---

C. Niloor · R. Agarwal

REVA Academy for Corporate Excellence, REVA University, Bangalore, India

e-mail: [chetan.ba04@рева.edu.in](mailto:chetan.ba04@рева.edu.in)

R. Agarwal

e-mail: [rashmi.agarwal@рева.edu.in](mailto:rashmi.agarwal@рева.edu.in)

P. Mishra

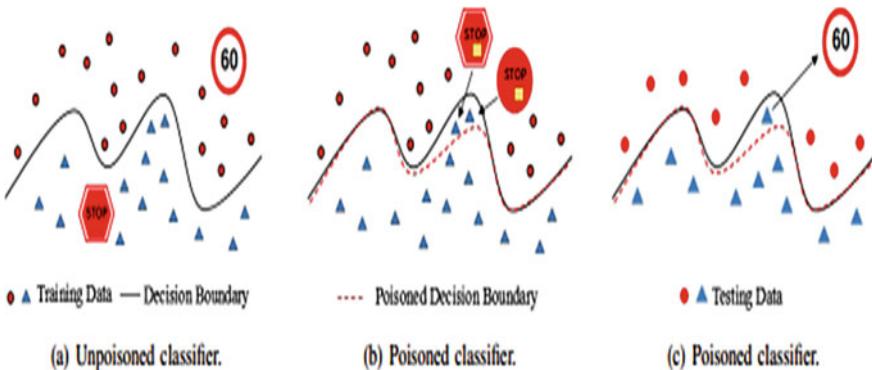
Co-Founder and Chief Architect, Data Safeguard Inc., Bangalore, India

## 1 Introduction

ML has grown into compelling element of diverse systems and applications. Research has revealed that ML algorithms are susceptible to serious vulnerabilities, notwithstanding the outstanding validity of ML algorithms in various signals and decision-making tasks. For instance, in model extraction attacks, an attacker can steal the proprietary and secret data of ML models, and in model inversion attacks, they can obtain the private and sensitive data of the training dataset. The model targets on data poisoning, where attackers keep disrupting the ML practices by inserting a chunk of pernicious piece of the dataset into the training dataset. Such weakness will put people at risk from numerous security risks like computer vision, biometric identification and self-driving automobiles.

For example, an attacker could add a “stop” sign with specific tags in the training data to change the decision, so the traffic sign could miscalculate the “stop” as the “speed limit” during examination phase Fig. 1, which would likely source self-driving cars to continue moving without stopping for any prevention [1].

To face poisoning attack, few methods are researched lately. Nevertheless, these particular defences are predominantly attack-specific: these are built for 1 anticipated type of attack and they are seriously susceptible to other types of attacks, that's because they follow definite regulations and fundamentals. To fill this breach, we have come up with De-Pois, an attack-agnostic approach for defence against poisoning attacks [2]. De-Pois is crucial for training mimic models, whose goal is to replicate the activities of the target model. By comparing the forecast between the produced mimic model and the target model, it can clearly possible to distinguish between poisoned samples and clean when utilizing the constructed mimic model approach.



**Fig. 1** An example of a data poisoning assault. **a** The classifier successfully classified the training data. **b** Attacker tamper with the decision boundary by introducing poisoned samples into the training dataset. **c** As a result of the poisoning attack, some testing data was incorrectly categorized during the testing phase (the “stop” sign was mistakenly interpreted as the “speed limit”) [2]

## 2 Literature Review

ML-based systems are significantly being implemented in settings with high collective impact, inclusive of hate speech detection on social networks, autonomous driving, biometric-based applications and malware detection. In actual world scenario applications, a system's toughness is not only to noise, but also influences is predominant. With the increase in total of machine learning systems trained on data garnered from public and semi-public areas alike social networking sites, cooperative-edited forums and multimedia community services, where attackers plan by injecting training dataset to influence or reduce system action significantly [1].

With great power comes great responsibility, as various research shows, that systems based on NLP (Natural Language Processing) are susceptible to attacks. Mostly in over-constant settings (where the number of parameters is larger compared to training data sets), which is common in most cases, NLP is unusually sensitive and docile to its expected pattern due to strategic plan disturbance in the dataset. Such restrictions are due to relational hostile corruption in data at the time of training, generally termed as data poisoning. These attacks instant a difficult problem, mostly where an adversarial could affect any factor of the training data [2].

ML techniques are widely used in a variety of applications. However, they are probably not safe from data poisoning attacks, in which potential attackers could perturb the training measure by introducing a small amount of harmful data samples into the training data. The defence strategies for data poisoning assaults are usually attack-explicit: they are created to a certain attack and are there for helpless against other types of attacks. This is mainly because they adhere to predetermined rules and fundamental concepts [2].

Data poisoning defences after factual verification are determined by certificates against different types of attacks. We accept that our ways are designed to be used as a way to check defence in the design stage, rather than assuring execution of a deployed learning algorithm (our strategy needs to run on the clean dataset, which we apparently won't be able to access it during the deployment time). For example, if we want to have a sturdy defence for image classifiers, we can check the outcome against attacks on a certain number of known image datasets, in order to boost more strength in the toughness of the system that we have actually deployed [3]. As of now there are four types of attacks and they are defendable to only specific type of defence techniques. Below is the list type of attacks and their defence techniques implemented on them:

**TCL-Attack (Target Clean-Label Attack Poisoning):** With a TCL-attack, the adversary introduces a weakly polluted dataset into the training set, increasing the risk that the machine learning model would mistakenly identify a clean test dataset sample when it is actually reviewed. Deep-KNN is used to exclude harmful samples before they may do damage during testing, based on the assumption that unique contaminated data samples will have different feature representations than clean data samples [2].

**pGAN Attack (Poisoning GAN Attack):** The generator, discriminator and classifier components of a pGAN-attack each separate accurate input from poisoned input. The defence against the pGAN assault is Certified Defence (CD), which eliminates outliers above the value viable and then minimizes a margin loss to remaining data. Then, using data-dependent constraints on the testing loss, CD assesses certified defences so that samples poisoned by a pGAN-attack that are farther away from the true class centroids can be located [2].

**LF-Attack (Label Flipping Attack):** The method of label flipping is used to attack SVM. The LF-attack gains the capability of inverting label to input data that are classified with accuracy, hence increasing its attacking potential. The teaching assignment will be addressed by creating a bi-level expansion issue as part of DUTI's defensive strategy to prevent this assault. Given a small sample of trustworthy data, DUTI would learn to recognize potential fraud and send them to subject-matter expertise for additional analysis to spot outliers [2].

**R-Attack (Regression Attack):** The main focus of R-Attack is on poisoning assaults for linear regression. An optimization problem with two levels is used to model the poisoning assault. For black-box assaults, a statistically-based poisoning attack is also envisioned, in which the attackers could object to the target model in order to learn about statistical properties of training data samples. TRIM builds a regression model with poisoned data on a portion of sample data and iteratively evaluates it to guard against R-attacks. The subdivision of samples with the smallest continuation can be recognized as providing accurate data. From Table 1, the adaptability of De-Pois can be seen. It is able to defend against all the types of attacks, whereas the other defence technique is susceptible to only 1 or 2 types of attacks [2].

The De-Pois technique consists of three steps: the generation of synthetic data, the construction of a mimic model and the recognition of poisoned data. Following these steps, the model will be able to train and mimic the model function before imitating the practice of a target model trained with clean samples. With the use GAN, training data augmentation and the creation of mimic models will be accelerated. De-Pois can distinguish between tainted and clean data samples by examining the predicted discrepancies for mimic model and target model [2].

Objective of this study is to build defence technique by using De-Pois to tackle attackers where they corrupt the ML models by data poisoning in training dataset. De-Pois is the only popular means of protection for poisoning attacks. De-Pois offers

**Table 1** Typical defences against the four types of poisoning are summarized [2]

Attack/Defence	Deep- Knn [20]	CD [28]	DUTI [29]	TRIM [22]	Sever [30]	De-Pois
TCL-attack [31]	YES	NO	NO	NO	NO	YES
pGAN-attack [32]	NO	YES	NO	NO	NO	YES
LF-attack [23]	NO	YES	YES	NO	YES	YES
R-attack [22]	NO	NO	YES	YES	YES	YES

**Effective defense:** YES **Ineffective defense:** NO

defence without any knowledge of any Machine learning (ML) or kinds of poisoning, and it can applied to both classification and regression tasks.

The distribution of clean data is mapped to the latent space representation using cGAN, which enables the addition of meaningful training data. In order to accommodate the Wasserstein distance between augmented and clean data for the construction of a mimic model, the conditional WGAN-GP (Gradient Penalty) will be further deployed. The findings demonstrate that De-Pois, with an average accuracy and F1-score more than 0.90, is extremely effective and successful at spotting poisoned data for all 4 types of attacks [2].

The new optimization framework for poisoning attacks and a fast statistical attack requires minimal knowledge of the training process [14].

The De-Pois process is a common defence system which will work for all types of attacks. There are three processes to be followed in De-Pois which are listed. Synthetic Data Generation: Clean data sets are used to generate enough fake data. The De-Pois uses cGAN to produce data and creates an authenticator to manage data augmentation.

Mimic Model Construction: De-Pois then creates the conditional version of WGAN-GP to learn prediction using the enhanced training data before creating mimic model. Discriminator is regarded as our mimic model once the conditional WGAN-GP training is complete.

Poisoned Data Recognition: The model use a detection boundary in conjunction with the mimic model to identify poisoned samples from uncontaminated ones. The sample is regarded as contaminated if the output of the mimic model is below our detection threshold.

### 3 Business and Data Understanding

When a competent adversary disrupts the ML procedure by inserting malicious data into training dataset, it poses a significant threat to numerous security-critical fields, including autonomous vehicles and biometric identity recognition. For example, attackers could simply insert a stop sign with specific stickers to training data to change decision, causing the traffic sign classifier to misinterpret “stop” as “speed limit” during the testing phase, which could result in self-driving cars continuing to steer without stopping for obstacle avoidance. Multiple forms of attacks are also seen in data poisoning attacks, including TCL-Attack, pGAN-Attack, LF-Attack and R-Attack. These attacks result in serious problem while training the model just by injecting a small set of data samples, which will adversely impact any of these attacks, that will result in low accuracy and over fitting, resulting inability to recognize the nonlinear data and outlier’s values. Hence there is strong need to identify and eliminate these serious threats [2].

The Existing methods have the accuracy to defend it, and there are several methods to identify and block these attacks, like Deep-KNN (K-Nearest Neighbours), TRIM (Total Risk Integrated Methodology), CD (Change Detection), Sever, but these

existing methods are not robust and efficient in detecting complex threats. When these existing defence techniques are implemented, they are bound to one or two types attacks not to all other attacks, also the accuracy of detecting poisoned data is 30% less compared to De-Pois, now the De-Pois is the only new defence technique which incorporates to all the types of attacks [2].

De-Pois defence requires significantly less time for data production, mimic model construction and poisoned data recognition compared to other existing defence systems, both in terms of training time and detection time, notwithstanding the impact of data dimension on training time.

De-Pois uses the MNIST dataset to illustrate synthetic data synthesis, mimic model construction and poisoned data recognition. These datasets will include the training set of 60,000 images of handwritten digits and the testing set of 10,000 digits. The core black-and-white images from MNIST have been adjusted to match a  $20 \times 20$  pixel size while preserving their aspect ratio. As a result of the anti-aliasing approach utilized by the normalizing algorithm, the final images consist of greyscale levels. The images were centred in a  $28 \times 28$  image by calculating the centre of mass of the pixels and translating the image to place this point in the centre of the  $28 \times 28$  field.

In training dataset and test data set images are labelled from the values between 0 and 9. In training and test data, the labels are set and the Pixels are organized row-wise. Pixel values are from 0 to 255, i.e. 0 means background (white), 255 means foreground (black) [1] (Fig. 2).

The used data is labelled dataset which will pair images of handwritten numbers with the name of each number, and it can be used as the basis for learning and practicing how to develop, evaluate and train the De-Pois using the Generative Adversarial Networks (GAN) and CGAN technique.



**Fig. 2** Sample images from MNIST test dataset [1]

By helping CGAN with the distribution of clean data, the De-Pois technique enhances CGAN and enables the creation of sufficient training data from a small chunk of a reliable clean dataset. Model is eventually able to run and mimic model to identify poisoned data from testing samples by comparing disparities between output of mimic model and built detection.

## 4 Data Preparation and Data Modelling

De-Pois initially creates enough synthetic data with distribution similar to  $S_c$  (clean dataset), assuming that only  $S_t$  (trusted dataset) may be acquired in practise. In order to understand  $S_c$  distribution, model uses cGAN for data creation. It creates an authenticator to manage the data augmentation. However, creating synthetic data is not a routine process, therefore we must make use of CGAN and Authenticator [2].

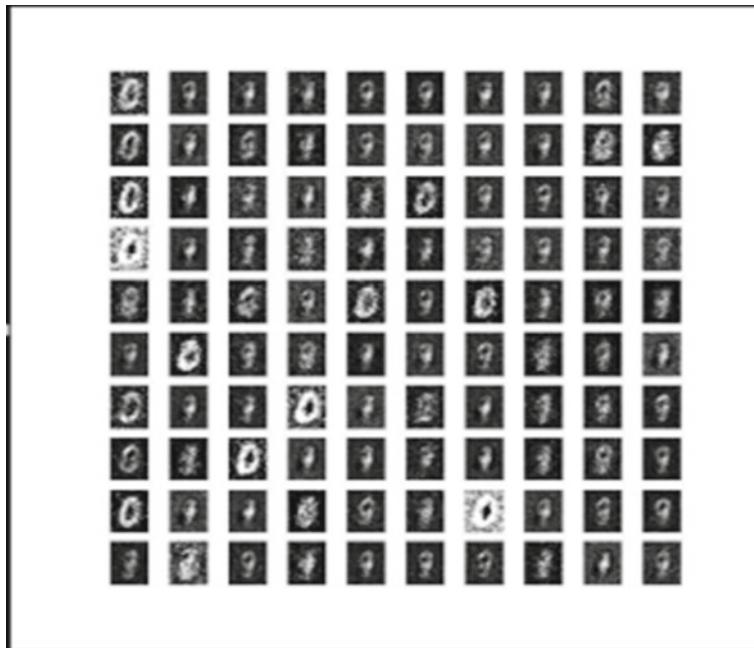
Conditional Generative Adversarial Networks (CGAN)-based generator: In initial part of CGAN, the D-discriminator takes the trusted and cleaned dataset as input which will learn to differentiate the clean datasets from other samples, simultaneously the G-generator learns to generate the synthesized samples by comprehending the distributions of training datasets.

Authenticator: Considering not to have any issues with augmented data (the trusted data and the synthetic data) which is used for training the mimic model and authenticator is used to administer the data augmentation process. During this procedure, the authenticator uses the freshly created samples at each repetition as an instance of missing variables that would lie in space of training dataset, ensuring that the generated data is trustworthy and distinct from each dataset [2]. The augmented data generation is done. The sample is shown in Fig. 3. From the synthetic data generation, it will be blur initially since it is getting trained while generating the data.

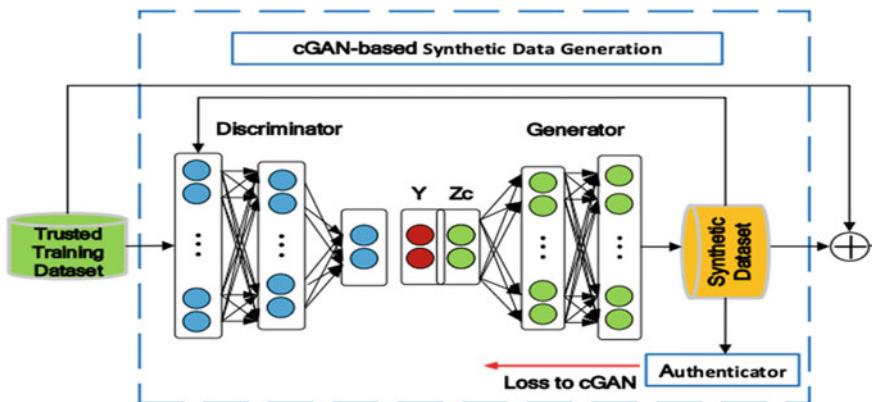
The Working of Synthetic Data Generation is demonstrated in Fig. 4, where trusted training dataset is generated using the generator and authenticator.

De-Pois is used to safeguard both classification and regression tasks, it attempts to ascertain, little quantity of (trusted) clean dataset. De-Pois is predicated on the idea that samples have been poisoned are more likely to have projections that are further away than samples with clean data. De-Pois therefore compares the predictions of the poisoned samples, using the mimic model with a cognitive prediction of target model developed by  $S_c$  [2]. A clean dataset  $S_c$  and a Poisoned dataset  $S_p$ .

**Synthetic Data Generation:** The synthetic data generating module typically consists of two components: a generator based on cGAN and an authenticator. The Generator(G) and the Discriminator neural networks are the two neural networks that make up the original GAN (D). While D learns to identify the real data samples  $x$  from G, G learns to create the synthesis samples  $G(z)$  that will recognize the distribution of training data from a prior noise distribution  $G(z)$ . The authenticator ensures a greater distinction between actual and created data, hence enhancing the data augmentation process [2].

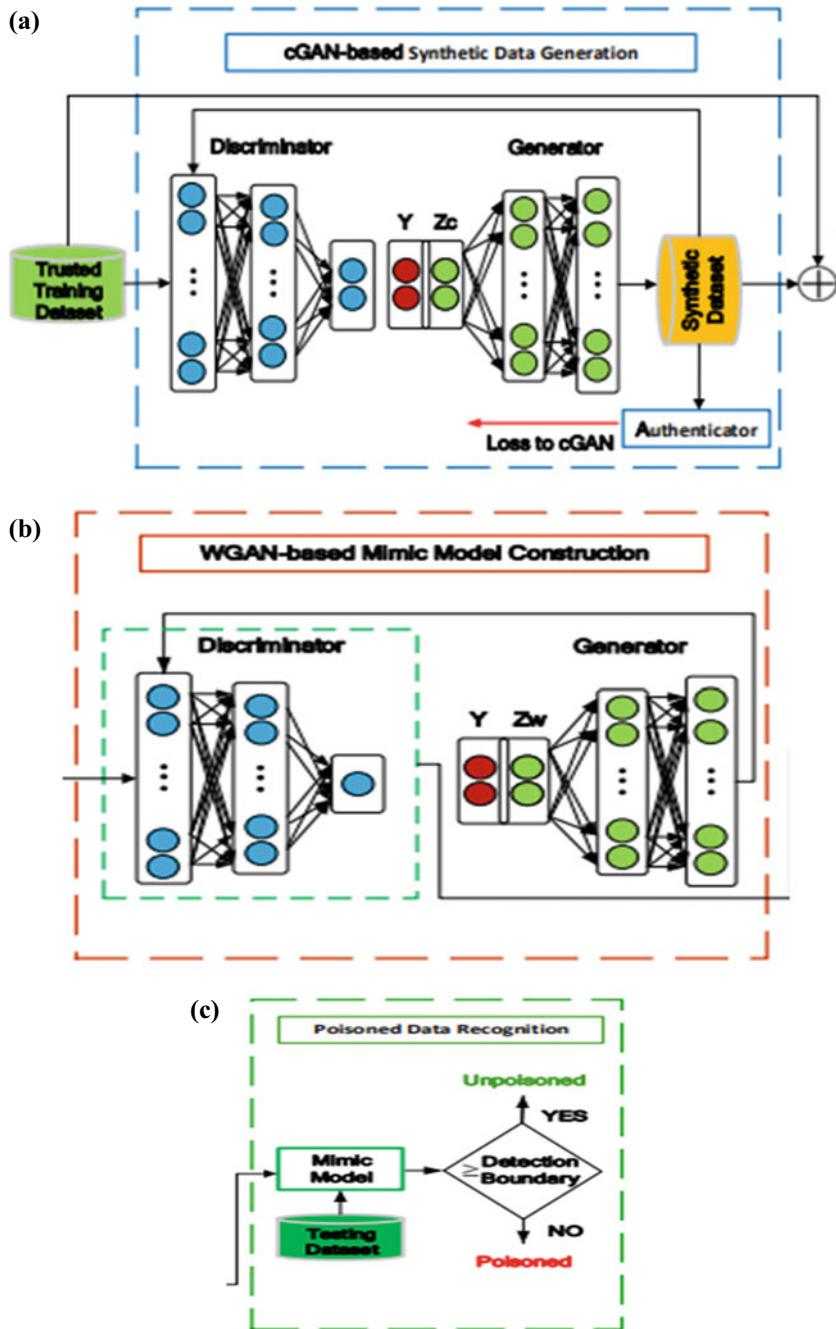


**Fig. 3** Augmented generation of data



**Fig. 4** Working of synthetic data generation [2]

**Poisoned Data Recognition:** A detection boundary is set up as demonstrated in Fig. 5a–c, by comparing the output of the mimic model to the detection threshold. If the output result is below our detection threshold, the sample is considered to be contaminated. Otherwise, the sample is non-poisoned [2].

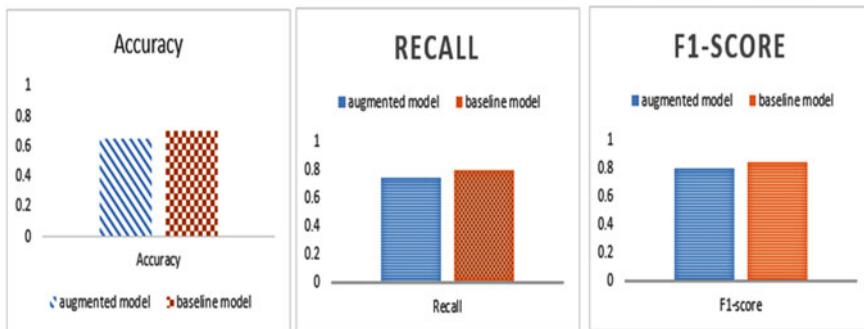


**Fig. 5 a–c** The architecture of implementation [2]

There are two steps in the cGAN training method for creating synthetic data. One is the discriminator component, where we want to reduce the LcGAN + LA by feeding it trusted clean samples ( $S_t$ ). In order to reduce LcGAN LA, a noise prior  $z_c$  is added information for the second generative component, where  $y$  is added as its input. These two elements are produced adversarially in synthetic data generation, as is typical in GAN. In order to get the cG framework [2]. From Fig. 6 In all three parameter ranges for accuracy, recall and F1, the success rate of synthetic data generation model design has increased by about 0.05, 0.03 and 0.04 respectively.

The Generated CGAN metric is shown in Table 2, where the IS score determines higher the score better is the result and if FID scores for real data, cGAN and De-Pois are lower the better is the result.

Every step in the De-Pois process has been evaluated. De-Pois will be correlated with three additional configurations, including cGAN+GAN, cGAN+cWGAN-GP and cGAN authenticator+GAN. Analysis of the results revealed accuracy of 20% for  $R_p$  and 20% for  $S_t$  and  $S_o$ . Figure 9. depicts the model's output; the authenticator model increases accuracy, recall and F1 by at least 0.03, 0.04 and 0.4, respectively. Finally, the range of all three parameters for accuracy, recall and F1 is increased by roughly 0.05, 0.04 and 0.03 thanks to the building of our mimic model [2].



**Fig. 6** Effectiveness of synthetic data generation

**Table 2** CGAN metric

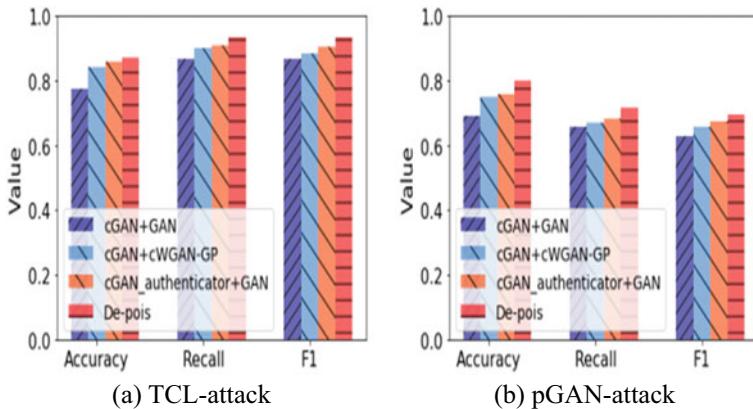
Dataset		MNIST	
Model		Metric	
		IS	FID
	Real Data	11.33	2.1
cGAN	6.1	51.03	
De-Pois	8.23	15.2	
<b>Inception Score (IS):</b> Higher is better			
<b>Frechet Inception Distance (FID):</b> Lower is better			

## 5 Deployment, Analysis and Results

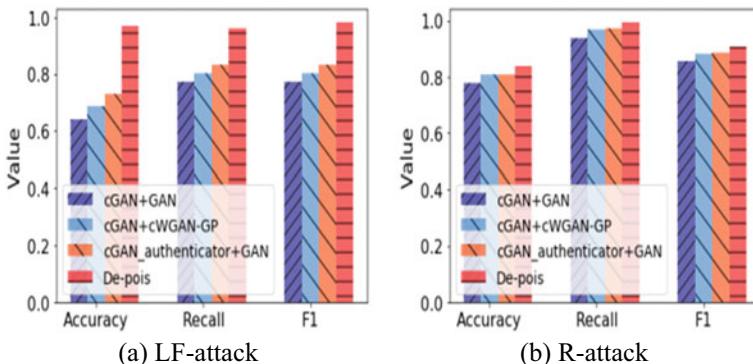
Three steps are involved in putting the De-Pois model into practise: creating synthetic data, building a model to imitate it and identifying poisoned data. In this process the developing the code so that it can be deployed in real-time applications in training the model to overcome the possessed threats. Figs. 7 and 8 show the outcome of De-Pois and simultaneously, zhelps to compare them with the other type of attacks to analyse the impact of De-Pois [2].

The time output for De-poise in comparison with other methods like CD, DUTI, TRIM and Sever is shown in Table 3.

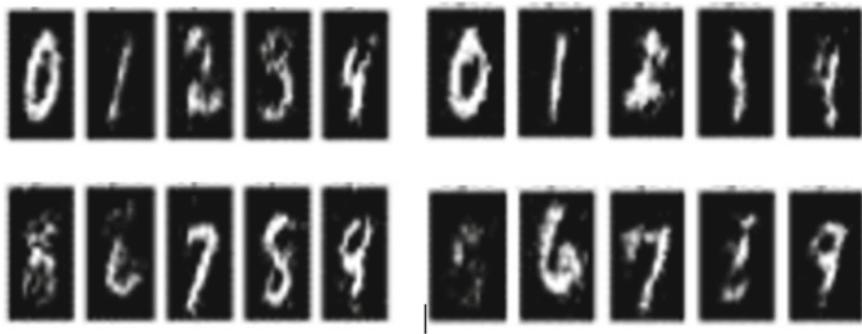
The analysis is done while implementing the De-Pois defence where we are generating the images to train and mimic the model. In Fig. 9 the De-Pois showcases the generation of synthetic data generation, which will look like below image. The clean



**Fig. 7** Impact of De-Pois in different types of attacks [2]



**Fig. 8** Impact of De-Pois in different types of attacks [2]



**Fig. 9** Synthetic data generation Image

**Table 3** De-Pois Comparison of detection time

Dataset	MNIST Dataset				
Method	De-Pois	CD	DUTI	TRIM	Sever
Detection time	0.108	0.05	0.069	0.007	0.027

data is generated with tags during the synthetic data generation. The images also show how the blur digits are cleaned slowly visible as the training of model progresses.

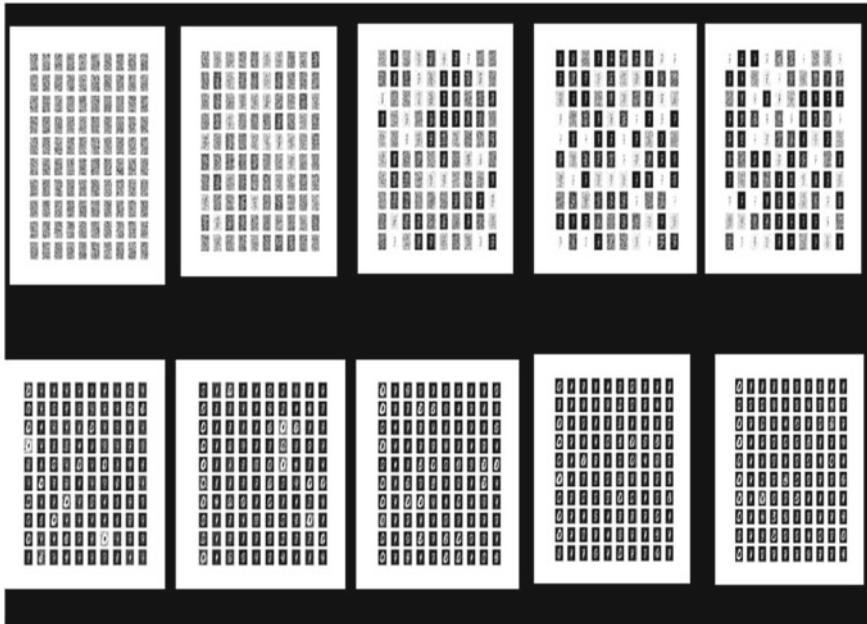
In Fig. 10, the generated synthetic data will be used for the poisoning by converting it into the augmented data, the augmented data can be seen here for reference.

The confusion matrix in Fig. 11 generated gives the result where the analysis of diagonal element, off-diagonal element is analysed for predicting and classifying the datasets:

1. The diagonal elements represent the number that are predicted.
2. Off-diagonal elements are those that are mislabeled by the classifier.
3. Higher the diagonal values of the matrix are considered as the good outcome.

From the generated confusion matrix, the diagonal values are higher than the off-diagonal values the model has predicted the correct values. The Accuracy in De-Pois is defined as the percentage of samples accurately predicted across all other predicting samples. In Fig. 12, the Accuracy generated in De-Pois is 0.822. The F1-Score is the weighted average of Precision and Recall value. Recall measures the ratio of precisely anticipated positive samples to all positive samples. De-Pois Re-Call value is 0.794%. Also, the greater the F1-Score, the better the De-Pois defence performs; the F1-Score output is 0.885.

**Conclusions and Future Scope:** De-Pois is the first solution of its kind designed to defend against all other types of data poisoning assaults. By supplying a modest amount of clean and trusted data, cGAN is adjusted to receive valid training data with similar distribution of clean data. Additionally, conditional WGAN-GP (Gradient Penalty) is used to train a successful mimic model that has comparable prediction



**Fig. 10** Augmented data

**Fig. 11** Confusion matrix

[[ 5019 117 95 97 101 88 105 105 99 97 ]]
[ 116 5803 116 109 98 90 111 120 90 89 ]
[ 92 123 5066 105 107 77 92 113 84 99 ]
[ 98 106 108 5210 102 85 114 103 99 106 ]
[ 117 104 85 103 4954 95 101 93 91 99 ]
[ 95 101 86 95 89 4596 78 97 77 107 ]
[ 91 116 112 109 77 88 5041 89 96 99 ]
[ 100 113 92 125 122 87 114 5297 119 96 ]
[ 98 107 107 101 97 85 110 119 4936 91 ]
[ 107 115 92 79 92 78 99 109 95 5083]]

capabilities to the target model. By finding the predicted discrepancies, we can distinguish between corrupted and clean data [2]. We expect that our effort will expand the understanding of data poisoning attacking/defending tactics and give insight on the development of more outlier identification techniques in a broader field.

**Recommendations:** De-Pois is attack-agnostic and effective defence technique, because it performs better than other known defense strategies, it can be evaluated against real-world data that is vulnerable to data poisoning. The use case of De-Pois

**Fig. 12** De-Pois findings

Accuracy of De-pois:  
0.8226181818181818  
Precision of De-pois:  
1.0  
Recall of De-pois:  
0.7946105263157894  
F1 of De-pois:  
0.8855520623152362

is more in Computer vision like autonomous driving, biometric identity recognition and computer vision, having significant societal effects, such as the identification of malware and hate speech on social media.

## References

- Chen J, Zhang X, Zhang R, Wang C, Liu L (2021a) De-Pois: an attack-agnostic defense against data poisoning attacks
- Anon (n.d.-a.) MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges. <http://yann.lecun.com/exdb/mnist/>
- Steinhardt J, Koh PW, Liang P (n.d.) Certified defenses for data poisoning attacks
- Tolpegin V, Truex S, Gursoy ME, Liu L (2020) Data poisoning attacks against federated learning systems
- Wang Y, Mianjy P, Arora R (2021) Robust learning for data poisoning attacks
- Tanner MA (2012) Tools for statistical inference: observed data and data augmentation methods. Springer
- Tramer F, Zhang F, Juels A, Reiter MK, Ristenpart T (2016) Stealing machine learning models via prediction APIs. In: Proceedings of USENIX security, pp 601–618
- Shafahi A, Huang WR, Najibi M, Suciu O, Studer C, Dumitras T, Goldstein T (2018) Poison frogs! targeted clean-label poisoning attacks on neural networks. In: Proceedings of NeurIPS, pp 6103–6113
- Paudice A, Muñoz-González L, Lupu EC (2018) Label sanitization against label flipping poisoning attacks, In: Proceedings of ECML PKDD, pp 5–15
- Udeshi S, Peng S, Woo G, Loh L, Rawshan L, Chattopadhyay S (1908) Model agnostic defence against backdoor attacks in machine learning. CoRR, arXiv 02203:2019
- Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: Proceedings of ICML, pp 214–223
- Biggio B, Roli F (2018) Wild patterns: ten years after the rise of adversarial machine learning. Pattern Recogn 84:317–331
- Wu J, Chen B, Luo W, Fang Y (2020) Audio steganography based on iterative adversarial attacks against convolutional neural networks. IEEE Trans Inf Forensics Secur 15:2282–2294
- Jagielski M, Oprea A, Biggio B, Liu C, Nita-Rotaru C, Li B (2018) Manipulating machine learning: poisoning attacks and countermeasures for regression learning. In: Proceedings of IEEE S&P, pp 19–35

15. Carnerero-Cano J, Muñoz-González L, Spencer P, Lupu EC (2020) Regularisation can mitigate poisoning attacks: a novel analysis based on multiobjective bilevel optimization. CoRR, arXiv: 2003.00040
16. Chen C, Zhao X, Stamm MC, Generative adversarial at-tacks against deep-learning-based camera model identification. IEEE Trans Inf Forensics Secur. <https://doi.org/10.1109/TIFS.2019.2945198>
17. Alfeld S, Zhu X, Barford P (2016) Data poisoning attacks against autoregressive models. In: Proceedings of AAAI, pp 1452–1458
18. Mei S, Zhu X (2015) Using machine teaching to identify optimal training-set attacks on machine learners. In Proceedings of AAAI, pp 2871–2877
19. Yang C, Wu Q, Li H, Chen Y (2017) Generative poisoning attack method against neural networks. CoRR, arXiv: 1703.01340
20. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In: Proceedings of NeurIPS, pp 6626–6637
21. Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB Swami A (2017) Practical black-box attacks against machine learning proceedings of the 2017 ACM on Asia conference on computer and communications security (ASIA CCS ‘17). ACM, New York, NY, USA, pp 506–519
22. Diakonikolas I, Kamath G, Kane D, Li J, Steinhardt J, Stewart A (2019) Sever: a robust meta-algorithm for stochastic optimization. In: International conference on machine learning, pp 1596–1606
23. Koh PW, Steinhardt J, Liang P (2018) Stronger data poisoning attacks break data sanitization defenses. arXiv preprint [arXiv:1811.00741](https://arxiv.org/abs/1811.00741)
24. Muñoz-González L, Lupu EC (2019) The security of machine learning systems. In: AI in cybersecurity. Springer, pp 47–79
25. Huang L, Joseph AD, Nelson B, Rubinstein B, Tygar JD (2011) Adversarial machine learning. 4th ACM workshop on artificial intelligence and security (AISeC 2011). IL, USA, Chicago, pp 43–57

# Direction Detection of Select Stocks with Machine Learning



Anand Mohan , Rashmi Agarwal , and J. B. Simha

**Abstract** Several research initiatives have been taken to predict stock market returns using historical data. Investors can find plenty of algorithms that detect the exact closing price of any stock but will not tell the direction of the closing price. During this proposed work, twenty-two years' price of the stock's daily close price is being utilized for direction detection. The objective of this paper is to get the right stock, perform exploratory data analysis for data preparation and then build the right models by using multiple modeling techniques to predict whether the price will move up or move down. Closing prices are being utilized as six different feature variables for building the classification model. The difference between the seventh and eighth day closing price is determined. The 0.7, 1, and 1.5% differences are different classes of direction to determine either positive, negative or no change. A similar process is again repeated for the feature variable increased to ten days and fourteen days respectively. Then momentum, trend, volatility, and volume indicators are utilized as feature variables and different classification models are built to determine upward direction detection. Random forest modeling has given the highest efficiency in direction detection. Logistic regression modeling done for percentage change in close price as 0.5% has given the highest efficiency for volume and momentum indicators whereas the extreme gradient boost classifier provided the best prediction performance for trend and volatility indicators. Therefore, various classification modeling techniques have been remarkably useful in direction detection for the stock under consideration.

**Keywords** Direction detection · Stock market · Technical indicators · Classification models · HDFC · KOTAK · SBI

---

A. Mohan ( ) · R. Agarwal

REVA Academy for Corporate Excellence (RACE), REVA University, Bangalore, India  
e-mail: [anand.ba06@reva.edu.in](mailto:anand.ba06@reva.edu.in)

J. B. Simha  
CTO, Abiba Systems, Bangalore, India

## 1 Introduction

The stock market encourages the free economy concept. It is one of the significant financial tools in the hands of the corporate and enterprises to raise their funds through investments done by the common man. In return for investors putting their stake in company stocks, it is expected that they earn profits through dividends and upward stock movements, which would also enhance their economic status apart from the growth of the participant company whose stocks are at stake in the public domain.

Live validations are still becoming a grim prospect, because of several things like value variations, quiet news, and existing noise [1]. Several machine learning associated techniques are developed which have created the potential to predict the market to an extent [2]. For the transaction of shares via a broker, there is mostly a fee paid to the broker for each buy and sale which will almost eat up the gains [3]. The requirement is to overcome the ambiguities of fundamental and technical evaluation, and advanced development in the modeling strategies has pushed several researchers to check new strategies for stock value forecasting [4].

In the next section, some of the available literature is scanned which throws light on various related aspects of machine learning methods and other methodologies, and also study and research other related issues which help assist better in direction detection in the stock market.

## 2 Literature Review

Literature review initially scans through technical and fundamental analysis of stocks. Further, it discusses as to how algorithmic trading based on fundamentals and technical indicators helps investors in their decision making. Further, it emphasizes the merits of machine learning and artificial intelligence over algorithmic trading. It discusses unsupervised and various supervised classification techniques used in this paper. Later it reviews the literature on confusion matrix discussing various metrics for evaluation of the modelling techniques used for this proposed work.

### 2.1 Fundamental and Technical Analysis

Rajkar et al. in their paper comprehensively talk about the numerous parameters impacting value movements in varied sizes and layers in the stock market [5]. Therefore, different analysis namely technical and fundamental analysis is being done to invest in stock markets.

Elbialy in his paper worked on fundamental analysis and suggested that it helps to identify and implement short positions by selling the shares of companies showing downtrends and then covering these positions by buying back the shares of these

companies when they start showing upward trends [6]. Fundamental analysis helps to identify stock quality and therefore, stock technical analysis done later performs better on the strong fundamental stock.

Thanekar and Shaikh in their paper conclude their study that technical analysis can demarcate and recognize commerce openings in the stock market by examining identifiable patterns similar to volume and price action movements [7]. Kimbonguil et al. in their paper used many technical indicators like Moving Average Convergence Divergence (MACD), moving average, etc. on the past costs to identify better stocks for trading purposes [8].

## 2.2 Algorithmic Trading

Taking the discussion further, Hansen in his paper mentions algorithmic trading which is a systematic method of trading without subjective assessment through a manual trader using computer programs. Hansen further observes that fast algorithms improve traders' ability to scan the market and seize opportunities most appropriately [9]. Mukerji et al. in the paper caution that though algorithmic trading gives better results than manual trading, regulators have restrained algorithmic trading following accusations of market manipulations [10].

## 2.3 Supervised and Unsupervised Learning

This paper introduces a new stock market prediction model that includes three major phases namely feature engineering, non-correlated feature selections, and finally direction detection.

Omta et al. in their paper used machine learning and artificial intelligence for the analysis of image-based cellular screens. It is suggested in the paper that exploratory data analysis should be performed as an initial step to gain a better data understanding before executing machine learning algorithms. Machine learning can again be further categorized into supervised and unsupervised learning [11].

Alhomadi findings in his paper were that some literature has used both supervised and unsupervised machine learning techniques for securities market predictive modeling and located that both kinds of models will create predictions with satisfactory accuracy (Acc) [12]. Dar researched further on unsupervised machine learning techniques by deeply studying principal component analysis and suggested that the central plan of principal component analysis is to spot correlations and patterns in a dataset with high dimensionality and scale back it to a considerably lower dimension without losing any important info [13].

Further, various supervised classification machine learning techniques have been used in this paper namely logistic regression, decision tree, random forest, k nearest neighbors, and extreme gradient boosting.

## 2.4 Classification Machine Learning Techniques

Al-Bairmani and Ismael worked further on exploring logistic regression and inferred that logistic regression is used instead of linear regression in situations where the target variable is not numeric, but a nominal or an ordinal variable [14]. Jena and Dehuri suggest that the simple linear modeling algorithms become more complex as the size of the datasets increases which is being handled using more advanced algorithms in the decision tree for classification and regression problems [15]. Schonlau and Zou infer that random forest modeling is quite flexible to non-linearity in the dataset and is the most appropriate ensemble learning algorithm for medium-sized to very large-sized datasets [16]. Wang studied k nearest neighbors and informs that it is the most popular statistical technique utilized in pattern identification over the last four decades [17]. Zhang et al. researched on extreme gradient boost which according to him is extensively recognized as an extremely useful ensemble learning algorithm. However, its performance needs more improvements ideally in scenarios where the dataset is imbalanced [18].

## 2.5 Confusion Matrix for Classification Models

Various classification algorithms as discussed have to be built for the data. Subsequently, all these algorithms have to be tested. The confusion matrix for classification models is a step in that direction. Markoulidakis et al. in their paper evaluate numerous performance metrics which include Acc, precision (Pr), and recall (Rcl) [19].

## 3 Methodology

Initially, fundamental and technical analysis of the stocks under consideration is performed to demonstrate why a particular stock dataset has been used for this proposed work. The Cross-Industry Standard Process for Data Mining (CRISP-DM) framework has been used in this paper. In data understanding, the different feature variables used for the proposed work are being studied and their univariate analysis is performed. Based on the data understanding phase, various steps are being taken in the data preparation phase namely handling missing values, features addition, and data scaling using the Minmax scaler. Once the data has been prepared, different modeling algorithms are implemented on them namely logistic regression, decision tree, random forest, k nearest neighbor, and extreme gradient boost classifiers. The data evaluation phase further examines the results of different modeling techniques which are used in the data modeling phase. Deployment speaks about developing a front end Application Programming Interface (API) for the deployment dashboard.

### 3.1 Data Collection

Various classification algorithms as discussed have to be built for the data. Subsequently, all these algorithms have to be tested. The confusion matrix for classification models is a step in that direction. Markoulidakis et al. in their paper evaluate numerous performance metrics which include Acc, Pr, and Rcl [19].

Daily trading data of HDFC, KOTAK, and SBI banks from the year 2000 to 2022 are being used for this study. This study uses National Stock Exchange (NSE) data.

The *symbol* column tells the corporate symbol mentioned for the stock. The *opening price* is the first trade worth that is recorded throughout the day's trading. The *high and low* is the highest and lowest value respectively at that a stock is listed during a period.

The *previous closing* is going to be a consecutive session's opening price. The *last price* is the one at which the foremost recent transaction happens. The close is the last value recorded once the market is closed on the day. The *Volume Weighted Average Price (VWAP)* is a trading benchmark based on both volume and worth. The trading *volume* shows the number of shares listed for the day, listed in lots of hundreds of quantities of shares.

Table 1 discusses details for every column used in the HDFC, KOTAK, and SBI datasets.

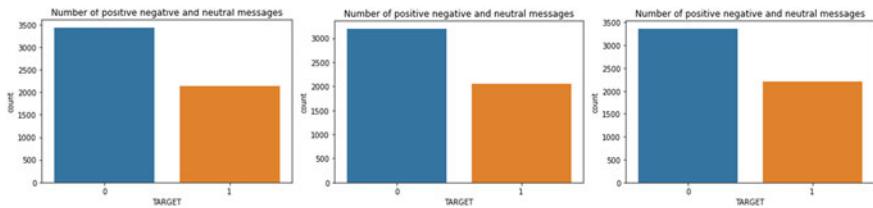
### 3.2 Data Exploration

As shown in Fig. 1, HDFC STOCK is moving 2140 times in an upward direction whereas 3435 times, it is not moving in an upward direction. KOTAK STOCK is 2055 times suitable for long trading whereas 3199 times, it is not moving in an upward direction. SBI STOCK is 2211 times suitable for long trading whereas 3364 times, it is not moving in an upward direction.

As shown in Figs. 2 and 3, the data has a positively skewed distribution which is observed in all 3 stocks namely HDFC, KOTAK, and SBI bank stock. SBIBANK stock is looking as the least volatile stock followed by HDFC and then KOTAK.

**Table 1** Top rows of HDFC, KOTAK, and SBI stock dataset

Date	Symbol	Prev close	Open	High	Low	Last	Close
1/3/2000	HDFC	272	294	294	294	294	294
5/30/2022	HDFC	2330	2368	2388	2362	2367	2367
1/3/2000	KOTAK	212	220	229	220	229	229
5/30/2022	KOTAK	1946	1945	1952	1896	1907	1903
1/3/2000	SBI	226	236	244	234	244	244
5/30/2022	SBI	469	473	477	471	475	475



**Fig. 1** Class Distribution for HDFC, KOTAK, and SBI stock

As shown in Fig. 4, a linear relationship exists between Independent variables and the target variable except for fewer outliers which is quite negligible.

### 3.3 Data Pre-processing

The HDFC, KOTAK, and SBI data which are taken from NSE come with a lot of limitations that have to be processed.

#### 3.3.1 Handling Missing Values

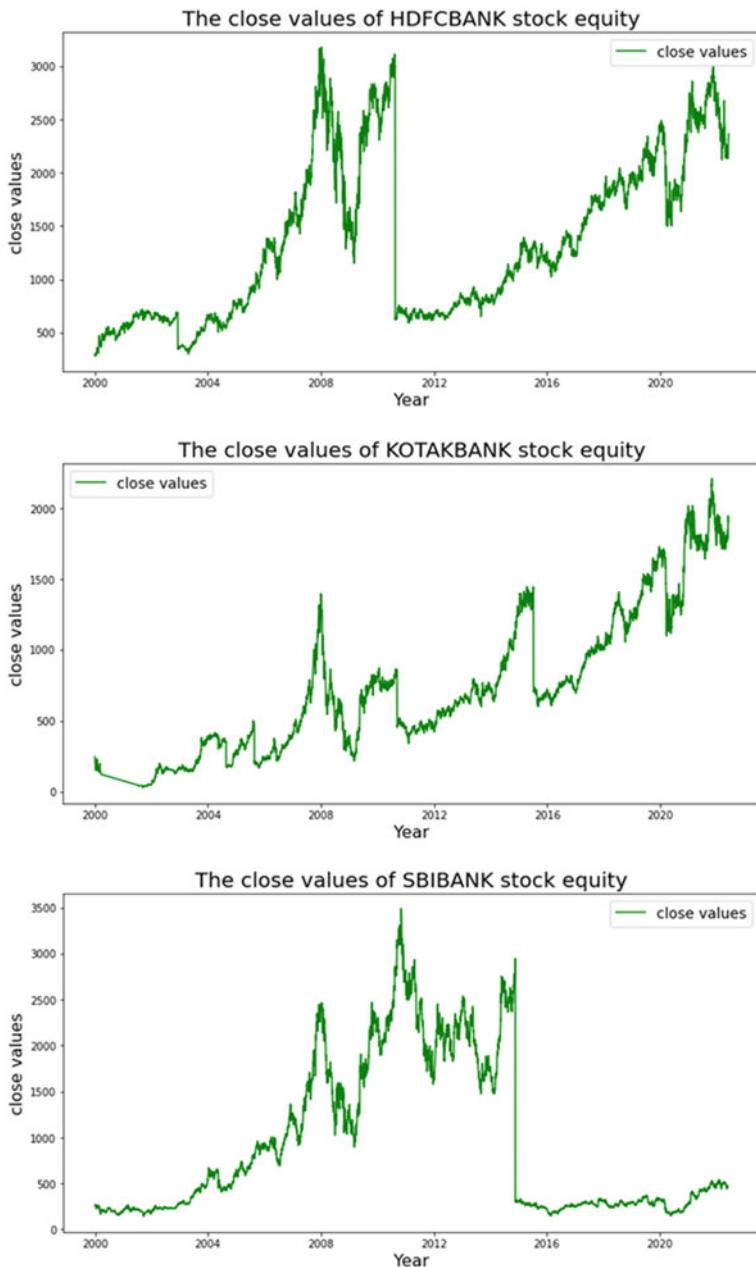
Three of the features' trades, 'deliverable volume', and '% deliverable' are dropped as they have several missing values.

#### 3.3.2 Features Addition

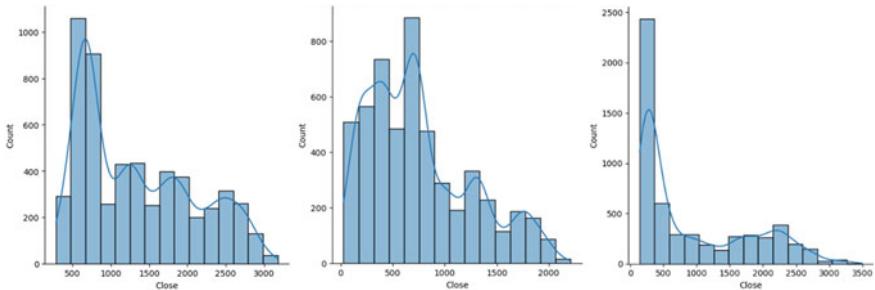
Computed variables added to the dataset are simple and exponential moving averages for rolling periods of seven, thirteen, twenty, hundred, and two hundred days. The one day's previous lag values of volume are also added as features. Six, ten, fourteen, and thirty days' consecutive closing prices are tabulated week on week for the entire dataset and utilized as different feature variables. Momentum, trend, volatility, and volume indicators are also used as feature variables.

#### 3.3.3 Data Scaling

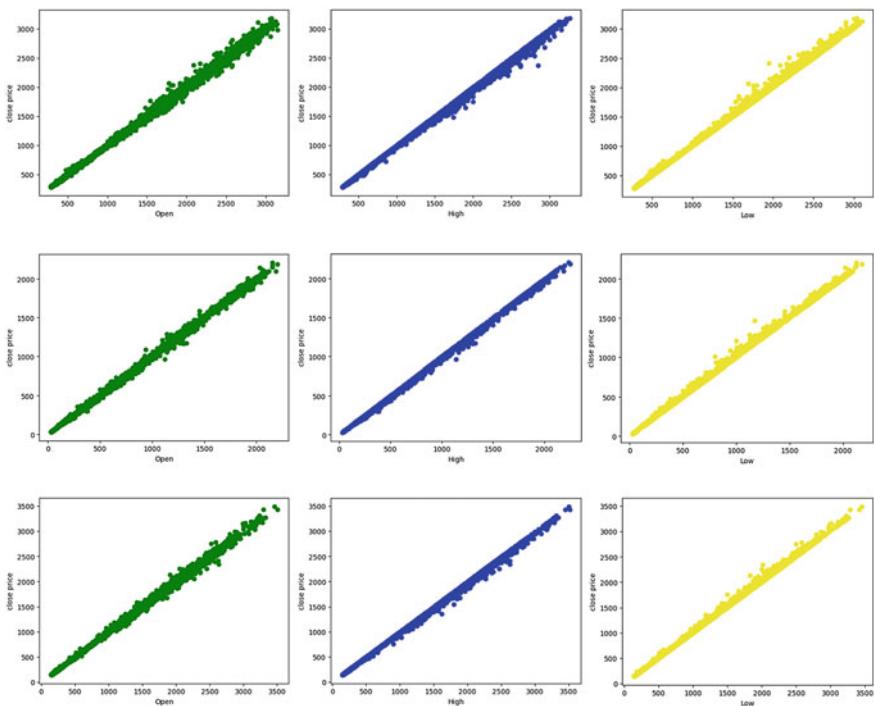
Minmax Scaler is the data scaling approach that is being used. MinMax Scaler shrinks the data inside the given range, from zero to one.



**Fig. 2** Close values of HDFC, KOTAK, and SBI stock from 2000 to 2022



**Fig. 3** Distribution Plot for the HDFC, KOTAK, and SBI Stock



**Fig. 4** Scatter Plot against close price for HDFC KOTAK, and SBI Stock from 2000 to 2022

### 3.4 Data Modeling

Based on direction detection Acc, it can be suggested to the prospective investor whether to invest or not invest in stock. Direction prediction Acc is further determined using momentum, trend, volatility, and volume indicators as feature variables and building different classification models on them. Table 2 explains the modeling strategies and model evaluation Rule used for this paper.

**Table 2** Modeling strategies and model evaluation rule

Modeling strategies	Model evaluation rule
Direction detection by six, ten, and fourteen days consecutive closing prices split week on the week	percentage change on closing price $> 0.7\% \Rightarrow$ Positive Trend percentage change on closing price $< -0.7\% \Rightarrow$ Negative Trend percentage change on closing price between 0.7 and 0.7% $\Rightarrow$ Neutral
Go long direction prediction is performed separately using momentum, trend, volatility, and volume indicators	percentage change on closing price $> 0.5\% \Rightarrow$ Positive Trend percentage change on closing price $\leq 0.5\% \Rightarrow$ Not Positive Trend

## 4 Implementation and Results

The data evaluation phase is the result of the data modeling phase and discusses the metrics utilized to determine the extent of the success achieved by the different modeling algorithms employed on the target variable.

### 4.1 Model Evaluation Using Logistic Regression Classifier for Go Long Direction Prediction

Various classification models are utilized to predict the direction of the close value of HDFC, KOTAK, and SBI stock and estimate using different error metrics. All the results derived from the various models are examined below.

From Table 3, it is observed that go long direction prediction using volume indicators has given considerable Pr, Rcl, and Acc in direction prediction.

### 4.2 Model Evaluation Using Random Forest Classifier for Go Long Direction Prediction

From Table 4, it is observed that direction detection has given the highest Pr, Acc, and Rcl in prediction. Also, go long direction prediction using volume indicators has given considerable Pr and Acc in direction prediction but Rcl can still be improved.

**Table 3** Model evaluation using logistic regression classifier

Modeling strategies	HDFC	KOTAK	SBI
Direction detection by six, ten, and fourteen days consecutive closing prices split week on week	Pr-0.35	Pr-0.37	Pr-0.36
	Rcl-0.60	Rcl-0.74	Rcl-1.00
	Acc-0.35	Acc-0.36	Acc-0.36
Go long direction prediction using volume indicators	Pr-0.98	Pr-0.99	Pr-0.92
	Rcl-0.83	Rcl-0.93	Rcl-0.80
	Acc-0.92	Acc-0.97	Acc-0.90
Go long direction prediction using momentum indicators	Pr-0.71	Pr-0.73	Pr-0.69
	Rcl-0.63	Rcl-0.61	Rcl-0.62
	Acc-0.76	Acc-0.75	Acc-0.74
Go long direction prediction using trend indicators	Pr-0.83	Pr-0.76	Pr-0.78
	Rcl-0.59	Rcl-0.48	Rcl-0.49
	Acc-0.80	Acc-0.72	Acc-0.74
Go long direction prediction using volatility indicators	Pr-0.93	Pr-0.90	Pr-0.81
	Rcl-0.47	Rcl-0.40	Rcl-0.30
	Acc-0.77	Acc-0.74	Acc-0.70

**Table 4** Model evaluation using random forest classifier

Modeling strategies	HDFC	KOTAK	SBI
Direction detection by six, ten, and fourteen days consecutive closing prices split week on the week	Pr-0.85	Pr-0.71	Pr-0.83
	Rcl-0.89	Rcl-0.79	Rcl-0.88
	Acc-0.87	Acc-0.74	Acc-0.85
Go long direction prediction using volume Indicators	Pr-0.91	Pr-0.92	Pr-0.90
	Rcl-0.82	Rcl-0.79	Rcl-0.73
	Acc-0.90	Acc-0.89	Acc-0.86
Go long direction prediction using momentum indicators	Pr-0.76	Pr-0.79	Pr-0.72
	Rcl-0.51	Rcl-0.46	Rcl-0.55
	Acc-0.75	Acc-0.74	Acc-0.74
Go long direction prediction using trend indicators	Pr-0.87	Pr-0.87	Pr-0.83
	Rcl-0.56	Rcl-0.55	Rcl-0.57
	Acc-0.80	Acc-0.79	Acc-0.78
Go long direction prediction using volatility indicators	Pr-0.89	Pr-0.89	Pr-0.83
	Rcl-0.50	Rcl-0.50	Rcl-0.61
	Acc-0.77	Acc-0.78	Acc-0.80

**Table 5** Model evaluation using random forest classifier

Modeling strategies	HDFC	KOTAK	SBI
Direction detection by six, ten, and fourteen days consecutive closing prices split week on the week	Pr-0.35	Pr-0.38	Pr-0.38
	Rcl-0.42	Rcl-0.41	Rcl-0.47
	Acc-0.40	Acc-0.40	Acc-0.37
Go long direction prediction using volume indicators	Pr-0.90	Pr-0.92	Pr-0.88
	Rcl-0.73	Rcl-0.90	Rcl-0.82
	Acc-0.86	Acc-0.93	Acc-0.89
Go long direction prediction using momentum indicators	Pr-0.70	Pr-0.75	Pr-0.70
	Rcl-0.61	Rcl-0.62	Rcl-0.59
	Acc-0.75	Acc-0.77	Acc-0.74
Go long direction prediction using trend indicators	Pr-0.85	Pr-0.82	Pr-0.83
	Rcl-0.74	Rcl-0.61	Rcl-0.67
	Acc-0.85	Acc-0.79	Acc-0.81
Go long direction prediction using volatility indicators	Pr-0.86	Pr-0.81	Pr-0.80
	Rcl-0.75	Rcl-0.63	Rcl-0.67
	Acc-0.85	Acc-0.79	Acc-0.81

### **4.3 Model Evaluation Using Extreme Gradient Boost Classifier for Go Long Direction Prediction**

From Table 5, it is observed that go long direction prediction using volume indicators has given considerable Pr, Rcl, and Acc in direction prediction.

### **4.4 Direction Detection and Go Long Direction Prediction Using the Best Classifier Model**

From Table 6, it is observed that random forest classifier modeling has given the highest efficiency in direction detection among all modeling techniques namely logistic regression, decision tree, random forest, k nearest neighbor, and extreme gradient boost modeling. This has been tested and proven with six, ten, and fourteen day consecutive closing prices split week on week as six, ten, and fourteen feature variables. Also, logistic regression classifier modeling has provided the best Pr, Rcl, and Acc for go long direction prediction using volume indicators.

### **4.5 Utility from the Business Perspectives**

For a stop loss of 2.0 reward-risk ratio for approximately 0.8 Pr would be  $2*0.8/2*0.2 = 4:1$  if a 0.5% difference in consecutive day close price for any stock is only 2.0 for higher percentage difference reward to risk ratio would be higher.

**Table 6** Leader board comparison of metrics for direction detection and go long direction prediction using the best classifier model

Modeling strategies	HDFC	KOTAK	SBI
Direction detection by six, ten, and fourteen days consecutive closing prices split week on the week (random forest classifier)	Pr-0.85	Pr-0.71	Pr-0.83
	Rcl-0.89	Rcl-0.79	Rcl-0.88
	Acc-0.87	Acc-0.74	Acc-0.85
Go long direction prediction using volume indicators (logistic regression classifier)	Pr-0.98	Pr-0.99	Pr-0.92
	Rcl-0.83	Rcl-0.93	Rcl-0.80
	Acc-0.92	Acc-0.97	Acc-0.90
Go long direction prediction using momentum indicators (logistic regression classifier)	Pr-0.71	Pr-0.73	Pr-0.69
	Rcl-0.63	Rcl-0.61	Rcl-0.62
	Acc-0.76	Acc-0.75	Acc-0.74
Go long direction prediction using trend indicators (extreme gradient boost Classifier)	Pr-0.85	Pr-0.82	Pr-0.83
	Rcl-0.74	Rcl-0.61	Rcl-0.67
	Acc-0.85	Acc-0.79	Acc-0.81
Go long direction prediction using volatility indicators (extreme gradient boost classifier)	Pr-0.86	Pr-0.81	Pr-0.80
	Rcl-0.75	Rcl-0.63	Rcl-0.67
	Acc-0.85	Acc-0.79	Acc-0.81

Here, modelling algorithms provide the close price of HDFC BANK, KOTAK BANK, and SBI BANK Stock over twenty years with the train test split of 70%:30%. If we invest Rs.10000 for six years and roughly calculate profit with 0.5% change on close price with the highest Pr in detecting true positives then the following results are possible as per the formulae given in Eq. (1):

$$n = \text{number of days it is true positive}$$

$$m = \text{number of days it is false positive}$$

$$pr = \text{precision}$$

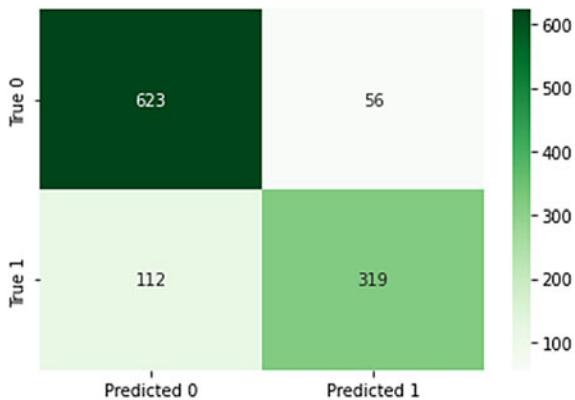
$$p = \text{percentage change}$$

$$c = \text{capitalinvest}$$

$$\text{netreturns} = (p * c * n * pr - p * c * m * pr) / 100 \quad (1)$$

Using trend indicators with the highest Pr of 0.85 for HDFC BANK stock, the confusion matrix provides information as shown in Fig. 5.

**Fig. 5** Confusion matrix for HDFCBANK stock using trend indicators as feature variables



Therefore, Net Returns are:

$$0.5 * 10000 * 319 * 0.85/100 - 0.5 * 10000 * 56 * 0.85/100$$

= Rs.11177.5 profit which would be

$$11177.5/(10000 * 6) * 100$$

= 18.63% returns

#### 4.6 Risk Adjusted Returns

The real data dump is imported for HDFC, KOTAK, and SBI stock between 2000 and 2022. Then the return, variance, and volatility of these stocks are calculated following which the annualized return to risk ratio and finally, the Sharpe ratios are calculated. The Sharpe ratio for HDFC, KOTAK, and SBI Stock is calculated as 0.173818, 0.149589, and 0.005306 respectively. Therefore, from the results obtained it becomes evident that HDFC shows a better Return vs. Risk performance over the specified period compared to KOTAK stock followed by the SBI stock which shows the least Return vs. Risk performance.

### 5 Conclusion and Future Scope

This paper solely focuses on predicting the direction of the close price of the HDFC stock using classification algorithms techniques. Later similar process is applied for predicting the direction of the close price of other stocks in the banking sector

namely SBI and KOTAK stocks. In the future, there will be a deployment dashboard proposed. As per the proposal for future assignments, the dashboard takes API as an input derived from the machine learning algorithms and can be utilized in predicting the direction of the close price for any stock in the banking sector. Any stock on the stock market can utilize the same procedure to forecast buy or not to buy choices, which is helpful.

This paper has not discussed how to address one major drawback of stock prediction, namely that over different periods the stock returns can change drastically. In future research work, it can be shown how to define bullish and bearish regimes using modern machine learning techniques. The sentiment analysis approach may also need to be explored using text analytics for predicting stock market returns. An intelligent automated system for options trading would be also the next step forward.

## References

- Shah D, Isah H, Zulkernine F (2019) Stock market analysis: a review and taxonomy of prediction techniques. *Int J Financ Stud* 7(2). <https://doi.org/10.3390/ijfs7020026>
- Sonkiya P, Bajpai V, Bansal A (2021) Stock price prediction using BERT and GAN [Online]. <http://arxiv.org/abs/2107.09055>
- Huang Y, Capretz LF, Ho D (2021) Machine learning for stock prediction based on fundamental analysis. In: Proceedings of the IEEE symposium series on computational intelligence (IEEE SSCI 2021). <https://doi.org/10.1109/SSCI50451.2021.9660134>
- Rouf N, et al (2021) Stock market prediction using machine learning techniques: a decade survey on methodologies, recent developments, and future directions. *Electron* 10(21). <https://doi.org/10.3390/electronics10212717>
- Rajkar A, Kumaria A, Raut A, Kulkarni N (2021) Stock market price prediction and analysis. *Int J Eng Res Technol* 10(06):115–119
- Elbialy BA (2019) The effect of using technical and fundamental analysis on the effectiveness of investment decisions of traders on the Egyptian stock exchange. *Int J Appl Eng Res* 14(24):4492–450 [Online]. <http://www.ripublication.com>
- Thanekar GS, Shaikh ZS (2021) Analysis and evaluation of technical indicators for prediction of stock market. *Int J Eng Res Technol* 10(May):341–344
- Kimbonguila A, Matos L, Petit J, Scher J, Nzikou J-M (2019) Effect of physical treatment on the physicochemical, rheological and functional properties of yam meal of the Cultivar ‘Ngumvu’ From Dioscorea Alata L. of Congo. *Int J Recent Sci Res* 10:30693–30695. <https://doi.org/10.24327/IJRSR>
- Hansen KB (2020) The virtue of simplicity: on machine learning models in algorithmic trading. *Big Data Soc* 7(1). <https://doi.org/10.1177/2053951720926558>
- Mukerji P, Chung C, Walsh T, Xiong B (2019) The impact of algorithmic trading in a simulated asset market. *J Risk Financ Manag* 12(2):68. <https://doi.org/10.3390/jrfm12020068>
- Omta WA et al (2020) Combining supervised and unsupervised machine learning methods for phenotypic functional genomics screening. *SLAS Discov.* 25(6):655–664. <https://doi.org/10.1177/2472555220919345>
- Alhomadi A (2021) Forecasting stock market prices: a machine learning approach. *Digit Commons* 11(2):16–36
- Dar AN (2021) Principal component analysis (PCA) (Using Eigen Decomposition). *Gsj* 9(7):240–252 [Online]. [www.globalscientificjournal.com](http://www.globalscientificjournal.com)

14. Al-Bairmani ZAA, Ismael AA (2021) Using logistic regression model to study the most important factors which affects diabetes for the elderly in the city of Hilla. *J Phys Conf Ser* 1:1818. <https://doi.org/10.1088/1742-6596/1818/1/012016>
15. Jena M, Dehuri S (2020) Decision tree for classification and regression: a state-of-the art review. *Inform* 44(4):405–420. <https://doi.org/10.31449/INF.V44I4.3023>
16. Schonlau M, Zou RY (2020) The random forest algorithm for statistical learning. *Stata J* 20(1):3–29. <https://doi.org/10.1177/1536867X20909688>
17. Wang L (2019) Research and implementation of machine learning classifier based on KNN. *IOP Conf Ser Mater Sci Eng* 677(5):5. <https://doi.org/10.1088/1757-899X/677/5/052038>
18. Zhang P, Jia Y, Shang Y (2022) Research and application of XGBoost in imbalanced data. *Int J Distrib Sens Netw* 18(6). <https://doi.org/10.1177/15501329221106935>
19. Markoulidakis I, Kopsiaftis G, Rallis I, Georgoulas I (2021) Multi-class confusion matrix reduction method and its application on net promoter score classification problem. *The ACM international conference proceeding series*. pp 412–419. <https://doi.org/10.1145/3453892.3461323>

# Prediction of Fatal Accidents in the Petrochemical Industry with Machine Learning



Thiyagarajan Balasubrani , Rashmi Agarwal ,  
and Mithun Dolthody Jayaprakash

**Abstract** There is an urgent need for energy companies to understand why employees still suffer injuries/accidents. The full potential of historical incident data is mainly unexplored and unexpressed. This study aims to question some preconceptions and identify opportunities for the future by better understanding the causes of catastrophic events, their effects, and the lessons that can be learned from them. In this paper, structured and unstructured data are taken to predict fatality. Structured data is analyzed with machine learning models and a combination of structured and unstructured data is analyzed with machine learning and deep learning models. For the structured data, out of all models, XGBoost gives an accuracy of 81%. And, for the combination of structured and unstructured data, Naïve Bayes with TF-IDF vectorizer gives an accuracy of 92%. Hence, this paper concludes that the Naïve Bayes with TF-IDF can be considered to predict the fatality of accidents with reasonable accuracy for a combination of structured and unstructured data.

**Keywords** Occupational accident · Naïve Bayes · XGBoost · Classification · Decision tree · Logistic regression · Deep learning (DL) · BERT · SVM

## 1 Introduction

Occupational accidents happen suddenly and unexpectedly. There have been incidents of workplace violence, fatalities, or accidents that have left workers there with personal injuries. According to one of Sarkar's (2016) important publications, there were roughly 335,000 fatal workplace accidents worldwide in 1994, or 14.0 fatal occupational accidents for every 100,000 workers on average. Due to the interruption in production, cost of medical care, labor compensation, and lost working time,

---

T. Balasubrani ( ) · R. Agarwal  
REVA Academy for Corporate Excellence, REVA University, Bengaluru, India  
e-mail: [thiyagarajanb.ba07@race.reva.edu.in](mailto:thiyagarajanb.ba07@race.reva.edu.in)

M. D. Jayaprakash  
Consultant, Statecraft, Bangalore, India

these accidents cause a loss in worldwide Gross Domestic Product (GDP) of close to 4%, or over US \$1.25 trillion [1]. Over 5700 persons in Europe lose their lives annually to workplace accidents [2]. In 2004, it was projected that the annual cost of accidents in the European Union (EU)-15 was 55 billion euros and that in the EU-27, 3.2% of workers, or 7 million people, had an accident at the workplace. Therefore, workplace accidents place a significant financial impact on the nation's economy as well as the sector [3].

Historical incidents are categorized according to the various causes and contributing elements. Additionally, machine learning models are created to forecast incident severity based on historical information such as varied event severity, including incident descriptions, incident kind, producing effect, and others [3]. The scope of the study is to prepare insights based on the historical incident structured and unstructured data, and the severity of the cases is predicted according to historical incident structured and unstructured data.

## 2 Literature Review

Bidirectional Encoder Representations from Transformers (BERT) and other transformer encoder architectures have had great success in Natural Language Processing (NLP) tasks. They generate natural language vector-space representations appropriate for deep learning (DL) algorithms [4]. BERT models are typically trained on a substantial corpus of text before being honed for purposes [5].

Numerous scholars employ various traditional methods like Bayesian networks, logistic regression, etc. Support vector machines (SVM) is one of the most frequently employed methods for classification out of all the methods discussed since it is effective at doing nonlinear classification. Although SVM performs well and has a high degree of accuracy based on the dataset, it cannot effectively distinguish between structured and unstructured data on its own. Most events and fatalities have been reported at oil and gas drilling sites in the Middle East and Southeast Asia, according to the general findings of this paper. Also, due to a lack of accident prevention, health and safety risk indicators in both drilling domains were substantially correlated with equipment maintenance. These incidents have been labeled as posing safety, chemical, ergonomic, and environmental threats [6].

Using NLP and ML, sentiment analysis seeks to ascertain if the predominant tone of texts is either positive or negative. BERT uses unsupervised language representation and bidirectional transformer models. BERT is capable of handling ambiguity in natural language processing. BERT can accurately evaluate languages that are like human languages [7].

Researchers agree that it is important to learn from past mistakes and advise a coordinated approach to lessen the risk of fire and explosion in the petrochemical industries [8].

Many machine learning tools are used to forecast hazardous situations involving toxic substances. Hazardous events are predicted using methods like logistic

regression, decision trees, neural networks, support vector machines, Naive Bayes classifiers, and random forests [9].

Preventative measures are produced for each of the incidents using the digitalized system, which is designed to facilitate the efficient reporting of incidents [10].

The causes of the disasters were examined using the data mining classification technique. Together with applying safety management measures, effective equipment usage and maintenance can prevent accidents [11].

The research proposes focused risk prediction and statistical analysis to solve the issue of the non-availability of the data. [12].

Business Intelligence tools are used to analyze the reported incidents. And further, the analysis is supported by clustering ML algorithms to group the incidents into manageable categories [13].

Several studies have been carried out to analyze and predict occupational fatalities by combining traditional Extraction Transformation and Loading (ETL) techniques and ML modeling techniques. Classic ML and DL techniques are carried out on structured data. A digitalized system to read real-time data was created to predict the fatalities. In this paper, the structured and unstructured data are merged, and classic ML and DL techniques are applied to predict the fatalities of the incident, which was not done in any other previous papers.

### 3 Material and Methods

Below is the list of algorithms used in this paper and their explanation is given.

SVM is one of the most popular and talked about machine learning algorithms. On a dataset, SVM may perform binary and multi-class classification [14, 15].

The logistic function can also be called as Sigmoid function. With the S-shaped curve, the real value number is converted to a value between 0 and 1 [16].

Decision Tree: A decision tree is a supervised learning technique that can be used to tackle classification and regression problems. It is a tree-structured classifier, where each leaf node represents the classification outcome and inside nodes represent the features of a dataset. The two nodes in a decision tree are the decision node and the leaf node [17].

Extreme Gradient Boosting (XGBoost): The Python XGBoost module implements gradient-boosted decision trees, which are the most important part of ML. XGBoost is a Python package that was developed by academics at the University of Washington. It is a C++-written Python package that aids ML model algorithms by providing training for gradient boosting [18].

Gradient Boosting: This Artificial Intelligence (AI) technique is used, among other things, in classification and regression assignments. It presents an expectation model as a group of mediocre decision trees for forecasting [18].

The main technological development of BERT is the unified architecture across multiple downstream tasks. It involves two steps, pre-training and fine-tuning [19].

TF-IDF stands for the Term Frequency Inverse Document Frequency of records. The frequency of a term in the corpus (dataset) offsets the effect of a word's frequency in the text, which raises meaning proportionately [20].

## 4 Proposed Model

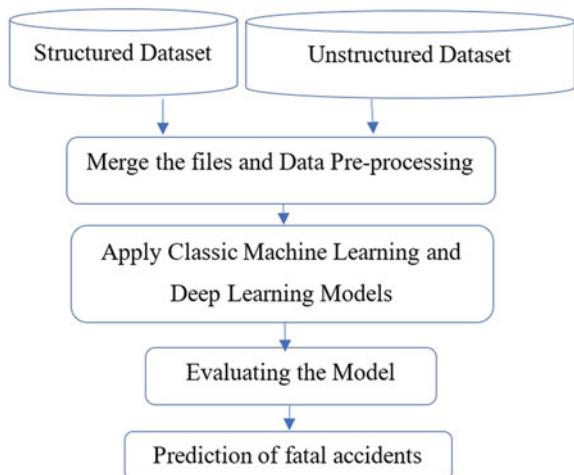
A total of 541 observations are taken into consideration from the incident management database. Data is cleansed and preprocessed before it is evaluated. The XGBoost model is considered for structured data and the Naïve Bayes model is considered for a combination of structured and unstructured data to predict fatal accidents. The methodological flow chart is given in Fig. 1. Finally, the model is deployed to predict fatal accidents.

## 5 Data Understanding

Occupational Incident data is collected from various sources. The list of Features considered is mentioned in Table 1. All the features are structured data except the field ‘Long Text’, which is unstructured. Portable Document Format (PDF) is read for each incident and attached to each incident for analysis and prediction.

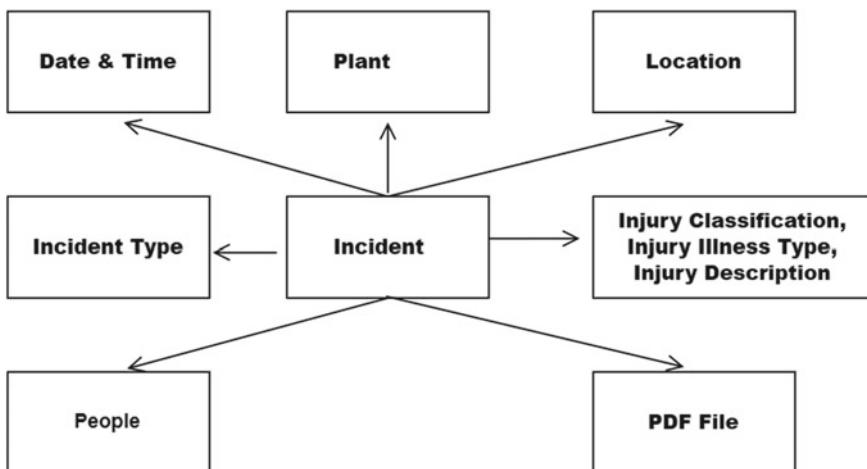
Figure 2 represents the attribute relationship of the data model. Incident holds the center stage of the data model. Date and time, plant, location, incident type, injury classification, injury illness type, injury description, people, and PDF file are related to the Incident with 1: N cardinality.

**Fig. 1** Proposed model

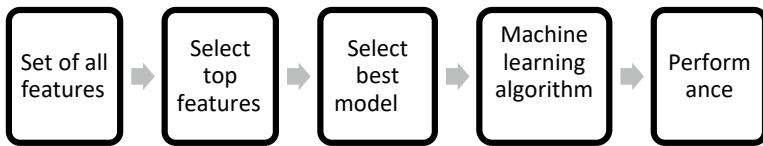


**Table 1** List of features

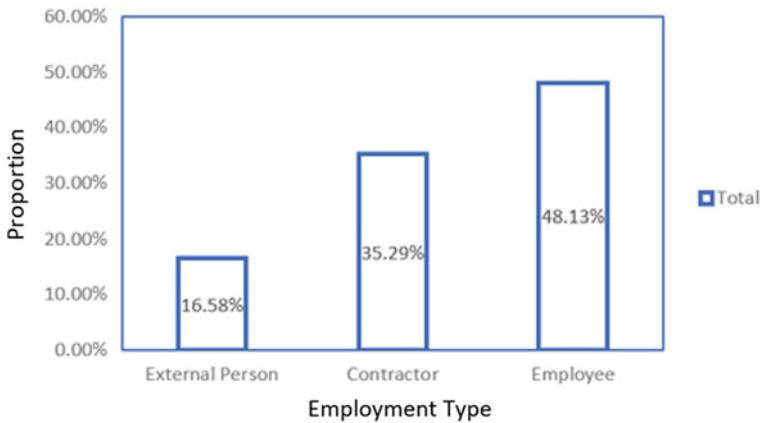
S. No	Name of the feature	Description of the feature
1	Incident creation date	Incident creation date
2	Incident creation time	Incident creation time
3	Plant	Independent, the physical organizational unit where manufacturing/refining takes place
4	Location	Location is a multi-level, hierarchical structure that is part of the technical structure
5	Causing effect	The cause of the incident occurred
6	Incident type	Type of the incident record
7	Injury classification	Specifies the type of injury the worker succumbs in the accident
8	Injury illness type	Specifies the injury/illness type
9	Injury description	Short text explanation about the accidents
10	Long text	PDF Attachment which describes the causes of the accident

**Fig. 2** Attribute relationship of the data model

The machine learning model learns well with a smaller number of features. The more features, the more searches the machine learning algorithm does, which reduces the performance. Feature selection in the machine learning pipeline is depicted in Fig. 3.



**Fig. 3** Feature selection in machine learning pipeline



**Fig. 4** Incident proportion by employment type

### 5.1 *Incident Proportion by Employment Type*

From Fig. 4, the people who suffered high severity incidents are 48.13% employees, 35.29% contractors, and 16.58% external persons. Of the total, permanent employees are affected more by high severity accidents. Followed by the contracts and external persons. This is understood that the company is not employing contract employees in the high-critical workplace.

### 5.2 *Incident Proportion by Incident Type*

From Fig. 5, ‘Community Accident’ and ‘Typical Accident’ are major types in the high severity incidents with 28.88% and 29.95% respectively.

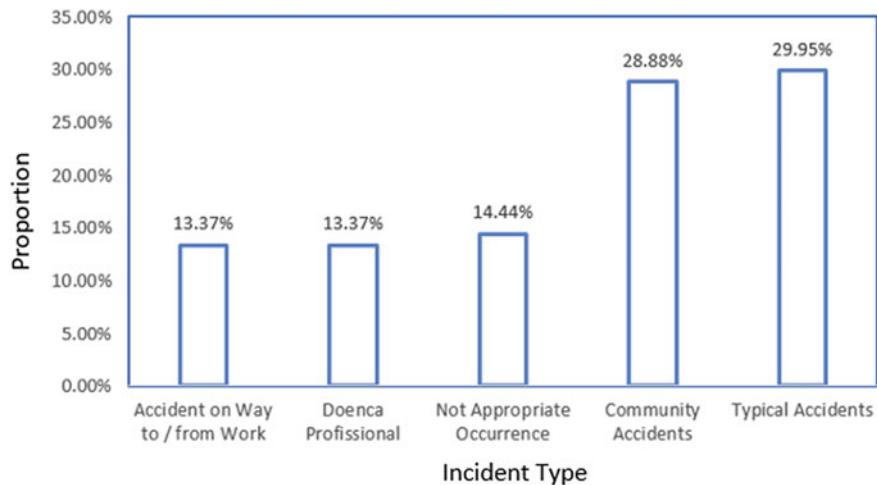


Fig. 5 Incident proportion by incident type

### 5.3 *Incident Proportion by Month*

From Fig. 6, in the month of 'June' and 'November' 21.39 and 18.72% of the high severity incidents occurred.

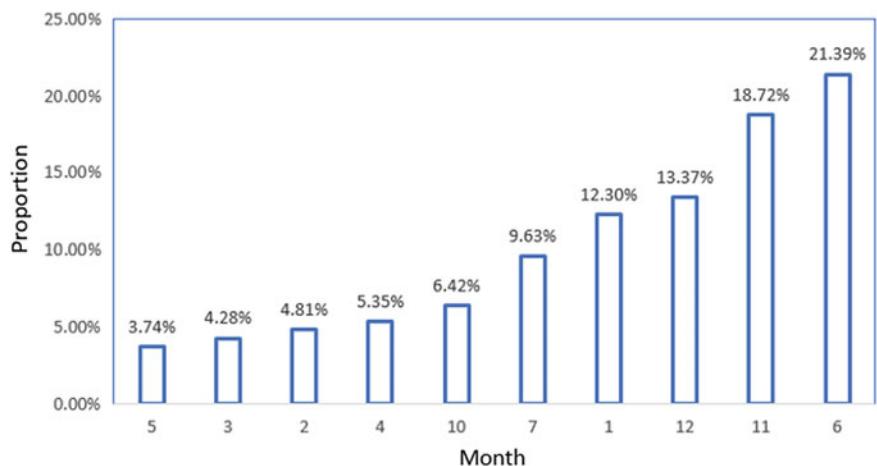
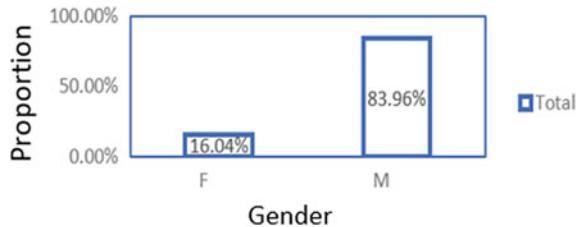


Fig. 6 Incident proportion by month

**Fig. 7** Incident proportion by gender



**Table 2** Incident proportion by causing effect

Incident causing effect	High severity (%)
Incident cause	25.67
Incident-causing object	21.39
Technical safety device	18.72
Immediate action	10.16
Near miss-causing substance	9.63
External influence	7.49
Movement of injured person	3.21
Personal protective equipment	2.14
Near miss-causing object	1.60
Total	100.00

#### 5.4 Incident Proportion by Gender

Figure 7 illustrates male employees constitute 83.96% of high severity incident cases, whereas 16% are female employees.

#### 5.5 Incident Proportion by Causing Effect

‘Incident Cause’ and ‘Incident-Causing Object’ constitute 25.67 and 21.39% of the high severity incidents, as shown in Table 2.

#### 5.6 Incident Proportion by Plant

Plant 1100 constitutes 33.69% and Plant 2800 constitutes 15.51% of high severity incidents, as shown in Table 3.

**Table 3** Incident proportion by plant

Plant	High severity (%)
1100	33.69
2800	15.51
4000	3.21
3600	3.21
2500	3.21
2700	3.21

**Table 4** Incident proportion by location

Location	High severity (%)
Fan Coil	5.35
Regulador	5.35
Vigesimo Quarto Pavimento—Ala sul	4.81
Garagem 02	2.67
Segundo Subsolo—Ala Leste	2.14
Indicador Ed Pressure Baixo	2.14
Vigesimo Quinto Pavimento—Ala norte	2.14
Pavimento 01	2.14
Segundo Subsolo—Ala Norte	2.14
Vigesimo Sexto Pavimento—Ala Leste	2.14

### 5.7 *Incident Proportion by Location*

Locations FAN COIL, Regulator and Pavimento 13 are the top affected locations, as shown in Table 4.

### 5.8 *Incident Proportion by Injury Illness Type*

‘Amputation’ constitutes 10.16% of high severity incidents, as shown in Table 5.

## 6 Implementation

The incident dataset contains tabular data, which is in a structured format. Support Vector Machines, Logistic Regression, Decision Tree, and XGBoost is used among classic ML models [21]. By using random sampling and replacement to create N new training data sets from the original dataset, certain observations may appear in more than one training dataset.

**Table 5** Incident proportion by illness type

Injury/Illness type	High severity (%)
Amputation	10.16
Multiple injuries (MOP)	2.67
Accidental poisoning by and exposure	2.14
Acute upper respiratory infections	2.14
Diseases of the genitourinary system	2.14
Scalping	1.60
Toxic effects of substances chiefly	1.60
Dislocation without fracture (FAT)	1.60

Along with the structured dataset, unstructured data in the form of PDF is also considered in this case. This represents a detailed description of incident-related data. For the combination of data, classic ML and DL models are considered. Naïve Bayes with TF-IDF vectorizer and BERT model is considered. By using the BERT model, numeric features in the dataset are converted to text [22]. Finally, text features are considered for prediction.

## 6.1 Structured Data

Multiple models are evaluated for the given dataset for predicting the incident severity. Support Vector Machines, Logistic Regression, Decision trees, and XGBoost are chosen among the classic ML models to predict the severity of the incidents.

### 6.1.1 Prediction Using XGBoost

The dataset was considered to meet the following criteria for applying XGBoost.

1. When training data contain a lot of observations.
2. The number of features in the training data is less than the number of observations.
3. It combines numerical and category characteristics.

The accuracy of the XGBoost classifier gives the highest accuracy of 81%, as shown in Fig. 8. Precision and recall for XGBoost are 79% and 95%, respectively. Receive Operating Characteristic Curve (ROC) for XGBoost is shown in Fig. 9.

SVM, Logistic Regression, and Decision Tree are also evaluated for the structured data.

Accuracy, Precision, and Recall for the selected models are given in Table 6.

```

1 xgb_classifier.fit(X_train,y_train)

XGBClassifier()

1 predictions = xgb_classifier.predict(X_test)

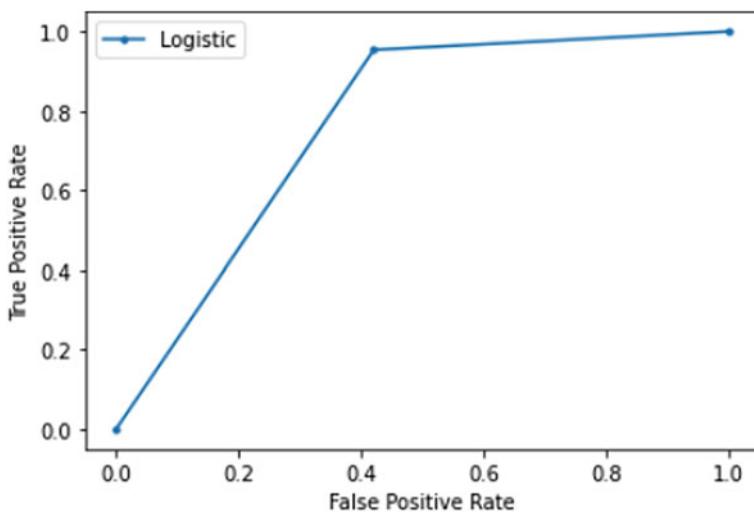
1 predictions

array([1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1,
       0, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1,
       1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])

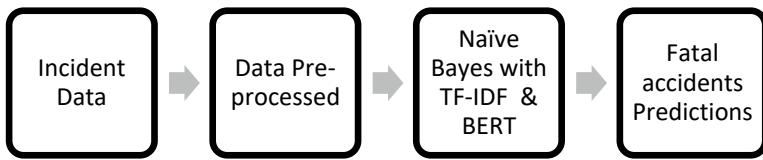
1 print("Accuracy of Model:::",accuracy_score(y_test,predictions))

Accuracy of Model::: 0.8155339805825242

```

**Fig. 8** Prediction using XGB classifier**Fig. 9** ROC curve for XGBoost**Table 6** Model evaluation for structured data

Metrics/Model	Accuracy	Precision	Recall
SVM	63.70	62.20	75
Logistic regression	64	64	80
Decision tree	75.50	74.10	95.40
XGBoost	81	79	95



**Fig. 10** Stages of data flow for prediction model

## 6.2 Combined Data

Structured and Unstructured data are combined, which is called as Combined data. In the case of combined data, BERT and Naïve Bayes model are taken into consideration.

The stages of data flow for the prediction model are illustrated in Fig. 10. Data is preprocessed and modeled with Naïve Bayes with TF-IDF and with BERT.

### 6.2.1 Prediction Using BERT Classifier

The BERT model is pre-trained with a single linear classification layer on top. The 12-layer BERT model is used to predict the severity. The number of output labels is 2 for binary classification, as illustrated in Fig. 11.

Training parameters for the BERT model are passed as batch size ‘32’, learning rate ‘1e-5’, and the number of epochs as ‘4’. The BERT model needs the number of epochs as less than or equal to 5 for text classification. The learning rate is the most important training parameter, which is set as ‘1e-5’. Training parameters are illustrated in Fig. 12.

Hyperparameters used in the BERT Model are given:

1. Optimizer: Adam
2. Number of Epochs: 4
3. Batch: 32
4. Learning Rate: 1e-5

Load the BERT Classification model.

```

1  from transformers import BertForSequenceClassification
2
3  # Load BertForSequenceClassification, the pretrained BERT model with a single
4  # linear classification layer on top.
5  model = BertForSequenceClassification.from_pretrained(
6      "bert-base-uncased", # Use the 12-layer BERT model, with an uncased vocab.
7      num_labels = 2, # The number of output labels--2 for binary classification.
8  )
9
10 # Tell pytorch to run this model on the GPU.
11 desc = model.cuda()
  
```

Downloading pytorch\_model.bin: 100% [420M/420M [00:07<00:00, 58.5MB/s]

**Fig. 11** BERT model initialization

## Training Parameters

```

1 # Larger batch sizes tend to be better, and we can fit this in memory.
2 batch_size = 32
3
4 # I used a smaller learning rate to combat over-fitting that I was seeing in the
5 # validation loss. I could probably try even smaller.
6 learning_rate = 1e-5
7
8 # Number of training epochs.
9 epochs = 4

```

**Fig. 12** BERT training parameters

Tokenizing the sentence by appending a token at the start of the sentence [CLS] and ending with a token [SEP]. The maximum sentence length is 442, as illustrated in Fig. 13.

Adam optimizer is used in this case. Adam is a class from the ‘huggingface’ library in which the Learning rate and epochs are passed as an argument to the optimizer, which is illustrated in Fig. 14.

The total number of training steps is calculated by the multiplication of the number of batches and the number of epochs. In this case, the number of batches is 32, and the number of epochs is 4. Hence, 128 training steps are carried out by the BERT model. The calculation of training steps and the creation of the learning rate scheduler are illustrated in Fig. 15.

The BERT model gives an accuracy of 63.1%.

### Get Maximum Length of the sentence

```

[ ] 1 max_len = 0
2
3 # For every sentence...
4 for sent in sen_w_feats:
5
6     # Tokenize the text and add '[CLS]' and '[SEP]' tokens.
7     input_ids = tokenizer.encode(sent, add_special_tokens=True)
8
9     # Update the maximum sentence length.
10    max_len = max(max_len, len(input_ids))
11
12 print('Max sentence length: ', max_len)

Max sentence length: 442

```

**Fig. 13** Input sentence length

## Create optimizer

```
[ ] 1 from transformers import AdamW
2
3 # Note: AdamW is a class from the huggingface library (as opposed to pytorch)
4 # I believe the 'W' stands for 'Weight Decay fix'
5 optimizer = AdamW(model.parameters(),
6                   lr = learning_rate,
7                   eps = 1e-8
8
9 )
```

**Fig. 14** Optimizer in BERT model

## \*\* learning rate scheduler will implement learning rate decay\*\*

```
1 from transformers import get_linear_schedule_with_warmup
2
3 # Total number of training steps is [number of batches] x [number of epochs].
4 # (Note that this is not the same as the number of training samples!)
5 total_steps = len(train_dataloader) * epochs
6
7 # Create the learning rate scheduler.
8 scheduler = get_linear_schedule_with_warmup(optimizer,
9                                               num_warmup_steps = 0, # Default value in run_glue.py
10                                              num_training_steps = total_steps)
```

**Fig. 15** Learning rate scheduler**6.2.2 Prediction Using Naïve Bayes with TF-IDF**

Text is preprocessed and used to calculate TF-IDF value based on the TF-IDF vectorizer function. Accuracy and AUC are calculated as shown in Figs. 16. and 17.

Table 7 illustrates that the Naïve Bayes with TF-IDF gives an accuracy of 92.02% and the BERT model gives an accuracy of 63%. The validation Loss curve for the BERT classifier is minimum with Epoch 4, which is shown in Fig. 18. However, the accuracy from the BERT model is lesser than the Naive Bayes. The ROC score for the Naïve Bayes model is 92%, as shown in Fig. 19 [23].

```

1  %%time
2  from sklearn.feature_extraction.text import TfidfVectorizer
3
4  # Preprocess text
5  X_train_preprocessed = np.array([text_preprocessing(text) for text in X_train])
6  X_val_preprocessed = np.array([text_preprocessing(text) for text in X_val])
7
8  # Calculate TF-IDF
9  tf_idf = TfidfVectorizer(ngram_range=(1, 3),
10                         binary=True,
11                         smooth_idf=False)
12 X_train_tfidf = tf_idf.fit_transform(X_train_preprocessed)
13 X_val_tfidf = tf_idf.transform(X_val_preprocessed)

CPU times: user 6.71 s, sys: 746 ms, total: 7.45 s
Wall time: 7.47 s

```

**Fig. 16** TF-IDF vectorizer

```

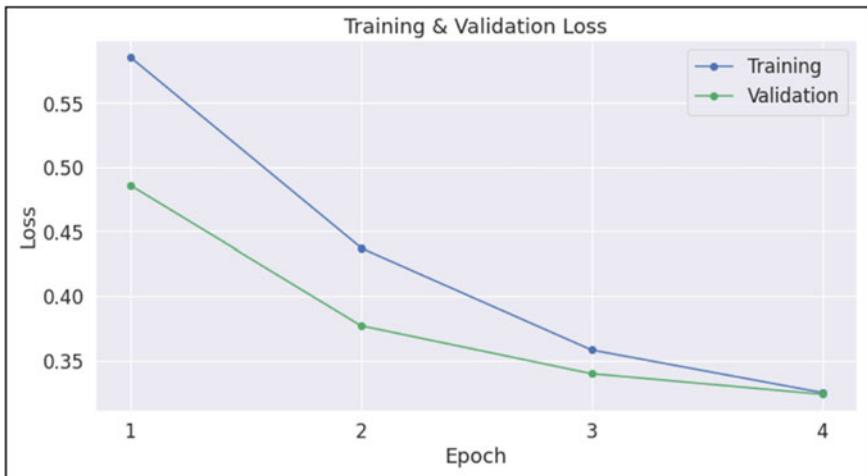
1  # Compute predicted probabilities
2  nb_model = MultinomialNB(alpha=1.8)
3  nb_model.fit(X_train_tfidf, y_train)
4  probs = nb_model.predict_proba(X_val_tfidf)
5
6  # Evaluate the classifier
7  evaluate_roc(probs, y_val)

AUC: 0.9202
Accuracy: 92.23%

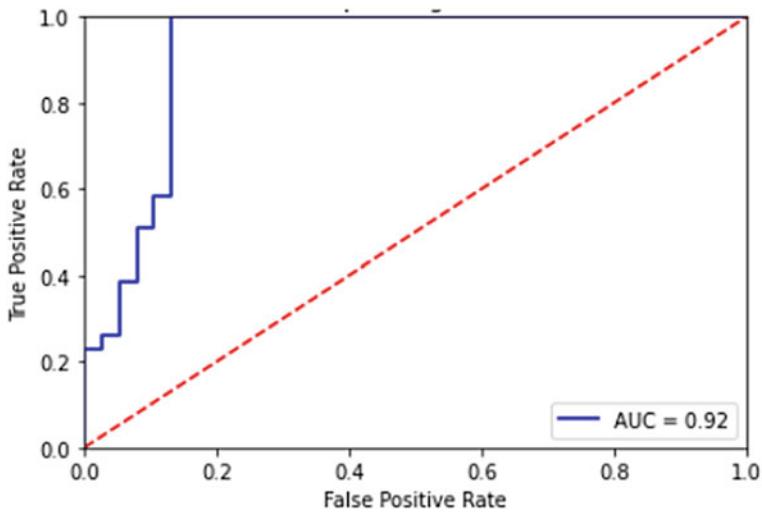
```

**Fig. 17** Evaluate using Naïve Bayes with TF-IDF classifier**Table 7** Model evaluation for combined data

Metrics/model	Accuracy (%)
Naïve Bayes with TF-IDF	92.02
BERT	63.10



**Fig. 18** Training and validation loss curve for BERT classifier



**Fig. 19** ROC curve for Naïve Bayes classifier

## 7 Conclusion

Structured and unstructured data are taken for the prediction of fatality in the petrochemical industry. Structured data is analyzed with machine learning models and a combination of structured and unstructured data is analyzed with machine learning and deep learning models. For the structured data, out of all models, XGBoost gives an accuracy of 81%. And, for the combination of structured and unstructured data,

Naïve Bayes with TF-IDF vectorizer gives an accuracy of 92%. Hence, this paper concludes that the Naïve Bayes with TF-IDF can be considered to predict the fatality of accidents taking place in the petrochemical industry. For the future scope, the model can be deployed in a cloud-based environment and a near real-time user interface can be developed to present the prediction results. Also, predictions can be made to identify the potential incidents converted into accidents.

## References

1. Sarkar S, Vinay S, Pateshwari V, Maiti J (2017) Study of optimized SVM for incident prediction of a steel plant in India. In: IEEE annual India conference INDICO2017N 2016. <https://doi.org/10.1109/INDICON.2016.7838894>
2. Deping L, Hongjuan W, Mengyang L, Pei L (2021) News text classification based on bidirectional encoder representation from transformers. In: Proceedings of The international conference on artificial intelligence and big data algorithms, CAIBDA. pp 137–140. <https://doi.org/10.1109/CAIBDA53561.2021.00036>
3. Flouri M, Karakosta C, Doukas H, Psarras J (2009) Review and analysis of oil and gas incidents related to the supply interruptions. In: ICEE 2009—Proceeding 2009 3rd International Conference on Energy Environment. Advanced Towards global sustainability. pp 171–176. <https://doi.org/10.1109/ICEENVIRON.2009.5398652>
4. Classify text with BERT,” TensorFlow, [Online]. [https://www.tensorflow.org/text/tutorials/classify\\_text\\_with\\_bert](https://www.tensorflow.org/text/tutorials/classify_text_with_bert). Accessed 20 Mar 2023)
5. JayAlammar. (n.d.). The Illustrated Transformer. Alammar J, The Illustrated Transformer. <https://jalammar.github.io/illustrated-transformer/>. Accessed 20 Mar 2023
6. Asad MM, Bin Hassan R, Sherwani F, Soomro QM, Sohu S, Lakhiar MT (2019) Oil and gas disasters and industrial hazards associated with drilling operation: an extensive literature review. In: 2019 2nd International conference on computing, mathematics and engineering technologies iCoMET. <https://doi.org/10.1109/ICoMET.2019.8673516>
7. Rothman D (2021) Transformers for Natural Language Processing Build innovative deep neural network architectures for NLP & Python, PyTorch, TensorFlow, BERT, RoBERTa & more. [Online]. <https://www.packtpub.com/product/transformers-for-natural-language-processing/9781800565791>
8. Loktionov OA, Kondrateva OE (2022) Development of a risk-based predictive model of employees fatal injuries in Russian power grid complex. In: Proceedings of 2022 4th international youth conference on radio electronics, electrical and power engineering. REEPE. pp 1–6. <https://doi.org/10.1109/REEPE53907.2022.9731480>
9. Chebila M (2021) Predicting the consequences of accidents involving dangerous substances using machine learning. Ecotoxicol Environ Saf 208:111470. <https://doi.org/10.1016/j.ecoenv.2020.111470>
10. Kurian D, Sattari F, Lefsrud L, Ma Y (2020) Using machine learning and keyword analysis to analyze incidents and reduce risk in oil sands operations. Saf Sci 130:104873. <https://doi.org/10.1016/j.ssci.2020.104873>
11. Cheng CW, Yao HQ, Wu TC (2013) Applying data mining techniques to analyze the causes of major occupational accidents in the petrochemical industry. J Loss Prev Process Ind 26(6):1269–1278. <https://doi.org/10.1016/j.jlp.2013.07.002>
12. Chu Z, You Z, Wang F, Chen X, Zhang B (2020) A targeted risk prediction method based on statistical analysis of accidents in petroleum geophysical exploration. J Pet Sci Eng 192(66):107314. <https://doi.org/10.1016/j.petrol.2020.107314>
13. Akel AJN, Patriarca R, Di Gravio G, Antonioni G, Paltrinieri N (2021) Investigating occupational and operational industrial safety data through business intelligence and machine learning. J Loss Prev Process Ind 73:104608. <https://doi.org/10.1016/j.jlp.2021.104608>

14. Brownlee J (2020) Support vector machines for machine learning. MachineLearningMastery.com, Aug., [Online]. <https://machinelearningmastery.com/support-vector-machines-for-machine-learning/>. Accessed 20 Mar 2023
15. 1.4. Support Vector Machines. Scikit-learn. <https://scikit-learn.org/stable/modules/svm.html>. Accessed 20 Mar 2023
16. Brownlee J (2020) Logistic regression for machine learning. MachineLearningMastery.com, Aug., [Online]. <https://machinelearningmastery.com/logistic-regression-for-machine-learning/>. Accessed 20 Mar 2023
17. Decision Tree Algorithm in Machine Learning—Javatpoint. [www.javatpoint.com](http://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm). <https://www.javatpoint.com/xgboost-ml-model-in-python>. Accessed 20 Mar 2023
18. XGBoost ML Model in Python—Javatpoint. [www.javatpoint.com](http://www.javatpoint.com). <https://www.javatpoint.com/xgboost-ml-model-in-python>. Accessed 20 Mar 2023
19. Horev R (2023) BERT Explained: State of the art language model for NLP,” *Medium*, Nov. 17, 2018. [Online]. <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>. Accessed 20 Mar
20. Understanding TF IDF Term Frequency Inverse Document Frequency (2023) Geeks-forGeeks. [Online]. <https://www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/>. Accessed 20 Mar 2023
21. Kumar UD (2017) Business analytics—the science of data-driven decision making, 1st edn. Wiley
22. Manzoor M et al (2021) RFCNN: Traffic accident severity prediction based on decision level fusion of machine and deep learning model. IEEE Access 9:128359–128371. <https://doi.org/10.1109/ACCESS.2021.3112546>
23. Sinha S, Jayan A, Kumar R (2022) An analysis and comparison of deep-learning techniques and hybrid model for sentiment analysis for movie review. In: 3rd International conference for emerging technology. INCET. pp 1–5. <https://doi.org/10.1109/INCET54531.2022.9824630>

# Author Index

## A

- Abhishek S. Masur, 49  
Abisha, D., 179  
Aditya, P. S., 149  
Anand Mohan, 229  
Ankit Gupta, 193, 201  
Ashutosh Kumar, 15  
Atrey Mahadev Anagal, 71  
Avinash More, 1  
Ayatullah Faruk Mollah, 117

## C

- Chetan Niloor, 213

## G

- Gueye, Bamba, 163

## H

- Harleen Kaur, 105

## J

- Jagathiraj, V. P., 89  
Jyoti Gupta, 193

## K

- Kalpana, N., 37  
Kanchan Vipul Bakade, 1  
Kiran Githala, 139  
Kiran More, 15

## M

- Malo, Sadouanouan, 163  
Manisha Gupta, 139  
Md. Yousuf Ansari, 117  
Meena, S. M., 49  
Midhun P. Mathew, 89  
Mithun Dolthody Jayaprakash, 245

## N

- Nupur Jangu, 61

## P

- Pradeepa Mishra, 213  
Praveen Devadakar, 71  
Praveen M. Dhulavvagol, 71

## R

- Rahul Srivastava, 193, 201  
Rajeshkanna Viswanathan, 149  
Rammurti Meena, 127  
Rashmi Agarwal, 213, 229, 245  
Ritu Chauhan, 105

## S

- Sachin Chauhan, 193, 201  
Sagar Kabadi, 149  
Samima Khatun, 117  
Satwik Kulkarni, 49  
Shankru Guggari, 49  
Shashank Agasimani, 149  
Shubhi Gupta, 201  
Simha, J. B., 229

Sitanshu Hallad, 49  
Sk. Saidul Alam, 117  
Sudheep Elayidom, 89  
Sudipta Chakraborty, 127  
Suji, S., 179  
Swaroop Anegundi, 71

**T**

Tanvi Bhujannavar, 49  
Thiyagarajan Balasubrani, 245

**U**

Uday Kulkarni, 49, 149

**V**

Vinayak Sudhakar Kone, 71  
Vipin Chandra Pal, 127  
Vittal Hundarad, 149

**Y**

Yati Varshney, 105

**Z**

Zahid Raza, 61  
Zougmore, Teegwende, 163