

Ashish Khanna · Deepak Gupta ·
Siddhartha Bhattacharyya ·
Aboul Ella Hassanien · Sameer Anand ·
Ajay Jaiswal *Editors*

International Conference on Innovative Computing and Communications

Proceedings of ICICC 2021, Volume 2

Advances in Intelligent Systems and Computing

Volume 1388

Series Editor

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences,
Warsaw, Poland

Advisory Editors

Nikhil R. Pal, Indian Statistical Institute, Kolkata, India

Rafael Bello Perez, Faculty of Mathematics, Physics and Computing,
Universidad Central de Las Villas, Santa Clara, Cuba

Emilio S. Corchado, University of Salamanca, Salamanca, Spain

Hani Hagras, School of Computer Science and Electronic Engineering,
University of Essex, Colchester, UK

László T. Kóczy, Department of Automation, Széchenyi István University,
Gyor, Hungary

Vladik Kreinovich, Department of Computer Science, University of Texas
at El Paso, El Paso, TX, USA

Chin-Teng Lin, Department of Electrical Engineering, National Chiao
Tung University, Hsinchu, Taiwan

Jie Lu, Faculty of Engineering and Information Technology,
University of Technology Sydney, Sydney, NSW, Australia

Patricia Melin, Graduate Program of Computer Science, Tijuana Institute
of Technology, Tijuana, Mexico

Nadia Nedjah, Department of Electronics Engineering, University of Rio de Janeiro,
Rio de Janeiro, Brazil

Ngoc Thanh Nguyen , Faculty of Computer Science and Management,
Wrocław University of Technology, Wrocław, Poland

Jun Wang, Department of Mechanical and Automation Engineering,
The Chinese University of Hong Kong, Shatin, Hong Kong

The series “Advances in Intelligent Systems and Computing” contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing such as: computational intelligence, soft computing including neural networks, fuzzy systems, evolutionary computing and the fusion of these paradigms, social intelligence, ambient intelligence, computational neuroscience, artificial life, virtual worlds and society, cognitive science and systems, Perception and Vision, DNA and immune based systems, self-organizing and adaptive systems, e-Learning and teaching, human-centered and human-centric computing, recommender systems, intelligent control, robotics and mechatronics including human-machine teaming, knowledge-based paradigms, learning paradigms, machine ethics, intelligent data analysis, knowledge management, intelligent agents, intelligent decision making and support, intelligent network security, trust management, interactive entertainment, Web intelligence and multimedia.

The publications within “Advances in Intelligent Systems and Computing” are primarily proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Indexed by DBLP, INSPEC, WTI Frankfurt eG, zbMATH, Japanese Science and Technology Agency (JST).

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

More information about this series at <http://www.springer.com/series/11156>

Ashish Khanna · Deepak Gupta ·
Siddhartha Bhattacharyya · Aboul Ella Hassanien ·
Sameer Anand · Ajay Jaiswal
Editors

International Conference on Innovative Computing and Communications

Proceedings of ICICC 2021, Volume 2



Springer

Editors

Ashish Khanna
Maharaja Agrasen Institute of Technology
Delhi, India

Siddhartha Bhattacharyya
Rajnagar Mahavidyalaya
Birbhum
India

Sameer Anand
Department of Computer Science
Shaheed Sukhdev College of Business
Studies
Rohini, India

Deepak Gupta
Department of Computer Science
Engineering
Maharaja Agrasen Institute of Technology
Rohini, Delhi, India

Aboul Ella Hassani
Faculty of Computers and Information
Cairo University
Giza, Egypt

Ajay Jaiswal
Department of Computer Science
Shaheed Sukhdev College of Business
Studies
Rohini, Delhi, India

ISSN 2194-5357

Advances in Intelligent Systems and Computing
ISBN 978-981-16-2596-1 ISBN 978-981-16-2597-8 (eBook)
<https://doi.org/10.1007/978-981-16-2597-8>

ISSN 2194-5365 (electronic)

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature
Singapore Pte Ltd. 2022

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,
Singapore

Dr. Ashish Khanna would like to dedicate this book to his mentors Dr. A. K. Singh and Dr. Abhishek Swaroop for their constant encouragement and guidance and his family members including his mother, wife and kids. He would also like to dedicate this work to his (Late) father Sh. R. C. Khanna with folded hands for his constant blessings.

Dr. Deepak Gupta would like to dedicate this book to his father Sh. R. K. Gupta, his mother Smt. Geeta Gupta for their constant encouragement, his family members including his wife, brothers, sisters, kids, and to my students close to my heart.

Prof. (Dr.) Siddhartha Bhattacharyya would like to Dedicated this book to Late Kalipada Mukherjee and Late Kamol Prova Mukherjee.

Prof. (Dr.) Aboul Ella Hassanien would like to dedicate this book to his wife Nazaha Hassan.

Dr. Sameer Anand would like to dedicate this book to his Dada Prof. D.C.Choudhary, his beloved wife Shivanee and his son Shashwat.

Dr. Ajay Jaiswal would like to dedicate this book to his father Late Prof. U. C. Jaiswal,

his mother Brajesh Jaiswal, his beloved wife Anjali, his daughter Prachii and his son Sakshaum.

ICICC-2021 Steering Committee Members

Patron(s):

Dr. Poonam Verma, Principal, SSCBS, University of Delhi.

Prof. Dr. Pradip Kumar Jain, Director, National Institute of Technology Patna, India.

General Chair(s):

Prof. Dr. Siddhartha Bhattacharyya, Christ University, Bangalore.

Prof. Valentina Emilia Balas, Aurel Vlaicu University of Arad, Romania.

Dr. Prabhat Kumar, National Institute of Technology Patna, India.

Honorary Chairs:

Prof. Dr. Janusz Kacprzyk, FIEEE, Polish Academy of Sciences, Poland.

Prof. Dr. Vaclav Snasel, Rector, VSB-Technical University of Ostrava, Czech Republic.

Conference Chair:

Prof. Dr. Aboul Ella Hassanien, Cairo University, Egypt.

Prof. Dr. Joel J P C Rodrigues, National Institute of Telecommunications (Inatel), Brazil.

Prof. Dr. R. K. Agrawal, Jawaharlal Nehru University, Delhi.

Technical Program Chair:

Prof. Dr. Victor Hugo C. de Albuquerque, Universidade de Fortaleza, Brazil.

Prof. Dr. A. K. Singh, National Institute of Technology, Kurukshetra

Prof. Dr. Anil K Ahlawat, KIET Group of Institutes, Ghaziabad.

Editorial Chair(s):

Prof. Dr. Abhishek Swaroop, Bhagwan Parshuram Institute of Technology, Delhi.

Dr. Arun Sharma, Indira Gandhi Delhi Technical University for Womens, Delhi.

Prerna Sharma, Maharaja Agrasen Institute of Technology (GGSIPU), New Delhi.

Convener:

Dr. Ajay Jaiswal, SSCBS, University of Delhi.

Dr. Sameer Anand, SSCBS, University of Delhi.

Dr. Ashish Khanna, Maharaja Agrasen Institute of Technology (GGSIPU), New Delhi.

Dr. Deepak Gupta, Maharaja Agrasen Institute of Technology (GGSIPU), New Delhi.

Dr. Gulshan Srivastava, National Institute of Technology Patna, India.

Publication Chair:

Prof. Dr. Neeraj Kumar, Thapar Institute of Engineering and Technology.

Dr. Hari Mohan Pandey, Edge Hill University, UK.

Dr. Sahil Garg, École de technologie supérieure, Université du Québec, Montreal, Canada.

Dr. Vicente García Díaz, University of Oviedo, Spain.

Publicity Chair:

Dr. M. Tanveer, Indian Institute of Technology, Indore, India.

Dr. Jafar A. Alzubi, Al-Balqa Applied University, Salt, Jordan.

Dr. Hamid Reza Boveiri, Sama College, IAU, Shoushtar Branch, Shoushtar, Iran.

Prof. Med Salim Bouhlel, Sfax University, Tunisia.

Co-Convenor:

Mr. Moolchand Sharma, Maharaja Agrasen Institute of Technology, India.

Organizing Chair(s):

Dr. Kumar Bijoy, SSCBS, University of Delhi.

Dr. Rishi Ranjan Sahay, SSCBS, University of Delhi.

Dr. Amrina Kausar, SSCBS, University of Delhi.

Dr. Abhishek Tandon, SSCBS, University of Delhi.

Organizing Team:

Dr. Gurjeet Kaur, SSCBS, University of Delhi.

Dr. Aditya Khamparia, Lovely Professional University, Punjab, India.

Dr. Abhimanyu Verma, SSCBS, University of Delhi.

Dr. Onkar Singh, SSCBS, University of Delhi.

Dr. Kalpana Sagar, KIET Group of Institutes, Ghaziabad.

Dr. Purnima Lala Mehta, Assistant Professor, IILM.

Dr. Suresh Chavhan, Vellore Institute of Technology, Vellore, India.

Dr. Mona Verma, SSCBS, University of Delhi.

Preface

We hereby are delighted to announce that Shaheed Sukhdev College of Business Studies, New Delhi, in association with National Institute of Technology Patna and University of Valladolid, Spain, has hosted the eagerly awaited and much coveted International Conference on Innovative Computing and Communication (ICICC-2021) in Hybrid Mode. The fourth version of the conference was able to attract a diverse range of engineering practitioners, academicians, scholars, and industry delegates, with the reception of abstracts including more than 3,600 authors from different parts of the world. The committee of professionals dedicated toward the conference is striving to achieve a high-quality technical program with tracks on Innovative Computing, Innovative Communication Network and Security, and Internet of Things. All the tracks chosen in the conference are interrelated and are famous among the present day research community. Therefore, a lot of research is happening in the above-mentioned tracks and their related sub-areas. As the name of the conference starts with the word “innovation”, it has targeted out-of-box ideas, methodologies, applications, expositions, surveys, and presentations helping to upgrade the current status of research. More than 900 full-length papers have been received, among which the contributions are focused on theoretical, computer simulation-based research, and laboratory-scale experiments. Among these manuscripts, 210 papers have been included in the Springer proceedings after a thorough two-stage review and editing process. All the manuscripts submitted to the ICICC-2021 were peer-reviewed by at least two independent reviewers, who were provided with a detailed review proforma. The comments from the reviewers were communicated to the authors, who incorporated the suggestions in their revised manuscripts. The recommendations from two reviewers were taken into consideration while selecting a manuscript for inclusion in the proceedings. The exhaustiveness of the review process is evident, given the large number of articles received addressing a wide range of research areas. The stringent review process ensured that each published manuscript met the rigorous academic and scientific standards. It is an exalting experience to finally see these elite contributions materialize into three book volumes as ICICC-2021 proceedings by Springer titled “International Conference on Innovative Computing and Communications”. The articles are organized into three volumes in

some broad categories covering subject matters on machine learning, data mining, big data, networks, soft computing, and cloud computing, although given the diverse areas of research reported it might not have been always possible.

ICICC-2021 invited seven keynote speakers, who are eminent researchers in the field of computer science and engineering, from different parts of the world. In addition to the plenary sessions on each day of the conference, ten concurrent technical sessions were held every day to assure the oral presentation of around 210 accepted papers. Keynote speakers and session chair(s) for each of the concurrent sessions had been leading researchers from the thematic area of the session. A technical exhibition was held on both days of the conference, displaying the latest technologies, expositions, ideas, and presentations. The research part of the conference was organized in a total of 28 special sessions and 3 international workshops. These special sessions and international workshops provided the opportunity for researchers conducting research in specific areas to present their results in a more focused environment.

An international conference of such magnitude and release of the ICICC-2021 proceedings by Springer has been the remarkable outcome of the untiring efforts of the entire organizing team. The success of an event undoubtedly involves the painstaking efforts of several contributors at different stages, dictated by their devotion and sincerity. Fortunately, since the beginning of its journey, ICICC-2021 has received support and contributions from every corner. We thank them all who have wished the best for ICICC-2021 and contributed by any means toward its success. The edited proceedings' volumes by Springer would not have been possible without the perseverance of all the steering, advisory, and technical program committee members.

All the contributing authors owe thanks to the organizers of ICICC-2021 for their interest and exceptional articles. We would also like to thank the authors of the papers for adhering to the time schedule and for incorporating the review comments. We wish to extend my heartfelt acknowledgment to the authors, peer-reviewers, committee members, and production staff whose diligent work put shape to the ICICC-2021 proceedings. We especially want to thank our dedicated team of peer-reviewers who volunteered for the arduous and tedious step of quality checking and critique on the submitted manuscripts. We wish to thank my faculty colleagues Mr. Moolchand Sharma and Ms. Prerna Sharma for extending their enormous assistance during the conference. The time spent by them and the midnight oil burnt is greatly appreciated, for which we will ever remain indebted. The management, faculties, administrative, and support staff of the college have always been extending their services whenever needed, for which we remain thankful to them.

Lastly, we would like to thank Springer for accepting our proposal for publishing the ICICC-2021 conference proceedings. Help received from Mr. Aninda Bose, the acquisition senior editor, in the process has been very useful.

Delhi, India

Organizers, ICICC-2021

Ashish Khanna
Deepak Gupta

Contents

Automatic Removal of Eye Blink Artefacts from EEG Data Using Spatio-Temporal Features	1
Rakesh Ranjan, A. Prabhakara Rao, and Anish Kumar Vishwakarma	
Autoencoder-Based Model for Detecting Accounting Statement Fraud	11
Hiral A. Patel, Abhishek Parikh, and Bhargav Patel	
Increase in Mental Health Cases Post COVID Outbreak	23
Agnideep Majumder, Mehardeep Singh Arora, Palak Mantri, and Ankur Saxena	
An Efficient Approach to Predict Fear of Human's Mind During COVID-19 Outbreaks Utilizing Data Mining Technique	41
Diti Roy, Tamal Joyti Roy, Iqbal Mahmud, and Nasif Alvi	
Evolutionary Algorithms for Face Recognition with Mask	53
Ekansh Chauhan, Manpreet Sirswal, Richa Singh, Nikhil Bagla, Bhaskar Kapoor, and Deepak Gupta	
Stock Price Prediction Using Reinforcement Learning	69
Poonam Rani, Jyoti Shokeen, Anshul Singh, Anmol Singh, Sharlin Kumar, and Naman Raghuvanshi	
Sentiment Analysis of Bangla Text Using Gated Recurrent Neural Network	77
Nasif Alvi, Kamrul Hasan Talukder, and Abdul Hasib Uddin	
Leveraging User Comments in Tweets for Rumor Detection	87
Shaswat Patel, Binil Shah, and Preeti Kaur	
A Cost-Efficient QCA XOR Function Based Arithmetic Logic Unit for Nanotechnology Applications	101
Divya Tripathi and Subodh Wairy	

Forecasting of PM10 Using Intelligent Crow Search Algorithm	117
Tuned Feed-Forward Neural Network	117
Shalini Shekhawat, Akash Saxena, A. K. Dwivedi, and Vishal Saxena	
A Hybrid Fusion-Based Algorithm for Underwater Image Enhancement Using Fog Aware Density Evaluator and Mean Saturation	129
Rosalind Margaret Paulson, Sruthi Gopalakrishnan, Sruthi Mahendiran, Varghese Paul Srambical, and Neethu Radha Gopan	
Application of Hybridized Whale Optimization for Protein Structure Prediction	141
Akash Saxena, Shalini Shekhawat, Ajay Sharma, Harish Sharma, and Rajesh Kumar	
Clinical Named Entity Recognition Methods: An Overview	151
Naveen S. Pagad and N. Pradeep	
Mobile Phone SMS Notification Behavior Analysis Using Machine Learning Technique	167
Sumaiya Deen Muhammad, Farzana Tasnim, and Sanjida Sharmin	
Computer Vision with Deep Learning Techniques for Neurodegenerative Diseases Analysis Using Neuroimaging: A Survey	179
Richa Vij and Sakshi Arora	
Breast Cancer Risk Prediction Using Different Clustering Techniques	191
Laboni Akter, M. Raihan, Md. Mohsin Sarker Raihan, Mounita Ghosh, Nasif Alvi, and Ferdib-Al-Islam	
Learner Model of Intelligent Tutoring System Based on Bayesian Network	205
Rohit B. Kaliwal and Santosh L. Deshpande	
CADBAIG: Context-Aware Dictionary-Based Automated Insight Generator	215
Shweta Taneja, Bhawna Suri, Praveen Arora, and Soumya Tanwar	
Human Depression Prediction Using Association Rule Mining Technique	223
Md. Al-Mamun Biilah, M. Raihan, Tamanna Akter, Nasif Alvi, Nusrat Jahan Bristy, and Hasin Rehana	
Implementation of A Smart Helmet with Alcohol and Fall Detection and Navigation System	239
Piyush Mishra, Pratik Pai, Pradhuman Singh, Vedant Kayande, and Manish Parmar	

BlockFITS: A Federated Data Augmentation Modelling for Blockchain-Based IoVT Systems	253
Bhrigu Kansra, Harshita Diddee, Tariq Hussain Sheikh, Ashish Khanna, Deepak Gupta, and Joel J. P. C. Rodrigues	
Comparative Analysis for Improving Accuracy of Image Classification Using Deep Learning Architectures	263
Gopal Sakarkar, Ketan Paithankar, Prateek Dutta, Gaurav Patil, Shivam, Ruchi Chaturvedi, Akshita Bhimarapu, and Riddhi Mandal	
Infrared Thermography-Based Facial Classification Using Machine Learning	275
Kumud Rani, Mala Kalra, and Rakesh Kumar	
An Efficient Cluster Assignment Algorithm for Scaling Support Vector Clustering	285
H. S. Jennath and S. Asharaf	
In Silico Analysis of Plant-Derived Medicinal Compounds Against Spike Protein of SARS-CoV-2 and Ace2	299
Tanya Sharma, Mohammad Nawaid Zaman, Shazia Rashid, and Seneha Santoshi	
Serverless Computation with NuLambda	315
R. Eashwaran, Tanya Mittal, and Kavita Sheoran	
Performance Comparison of Different Machine Learning Algorithms on Hindi News Classification	323
Monika Arora, Bhumika Dhingra, Dhruv Gupta, and Dajinder Singh	
ConvLSTM for Human Activity Recognition	335
Ramendra Singla, Shubham Mittal, Alok Jain, and Deepak Gupta	
Mathematical Scanner (M-Scan) Mobile Application for Solving Simple Math Equations	345
Mamta Mittal, Gopi Battineni, Waqar Ahmad, Nitin Kumar, and Ravi Upreti	
Usability Evaluation of Novel Text CAPTCHA Schemes Based on Colors and Shapes	355
Tejaswi Kumar, Navansh Goel, Siddhant Roy, and C. Oswald	
Feature Selection for Email Phishing Detection Using Machine Learning	365
Neelam Yadav and Supriya P. Panda	
Workflow Scheduling Using Optimization Algorithm in Fog Computing	379
Gaurav Goel, Rajeev Tiwari, Abhineet Anand, and Sumit Kumar	

Early Detection of Covid-19 Based on Preliminary Features Using Machine Learning Algorithms	391
Madhav Sharma, Ujjawal Prakash, Anshu Kumari, and Kanika Singla	
Detection of Green Belt Area Using Machine Learning Algorithms	403
Ashish Raj Mahato, Rakshit Luke Wilson, Sushmita Sahoo, and Kanika Singla	
Survey on Social Distancing Detection Using Deep Learning	415
Harsh Sandesara, Karan Shah, and Pramod Bide	
Skin Burn Detection Using Machine Learning	427
Ashish Sharma	
Video Event Classification and Recognition Using AI and DNN	435
Sandeep Rathor, Nitika Garg, Prateek Verma, and Sarthak Agrawal	
Systematic Survey on Cryptographic Methods Used for Key Management in Cloud Computing	445
Ramakrishna Oruganti and Prathamesh Churi	
Domain-Controlled Title Generation with Human Evaluation	461
Abdul Waheed, Muskan Goyal, Nimisha Mittal, and Deepak Gupta	
A Big Data Query Optimization Framework for Telecom Customer Churn Analysis	475
Aarti Chugh, Vivek Kumar Sharma, Manjot Kaur Bhatia, and Charu Jain	
Lung Cancer Detection in Radiographs Using Image Processing Techniques	485
Bhawan Deep Singh, Chakshu Sharma, and Ashish Khanna	
COVID-19 Spread: A Demographic Analysis	497
Yashi Srivastava, Pooja Khanna, Sachin Kumar, and Pragya	
A One-Dimensional CNN Model for Subject Independent Emotion Recognition Using EEG Signals	509
Pallavi Pandey and K. R. Seeja	
Classification and Diagnosis of Alzheimer's Disease from ADNI Dataset Using RBM Classifier	517
Simarjeet Singh and Rekh Ram Janghel	
A Comparative Study of Early Detection of Diabetes Risk by Machine Learning	531
Ishmeet Kaur Aubi, Swati Chauhan, and Sanjeev Kumar Prasad	
Monitoring of the COVID-19 Cases by EWMA Control Chart	541
Pulak Kumari, Anurag Priyadarshi, Amit Kumar Gupta, and Sanjeev Kumar Prasad	

Detecting the Trend of a Product by Online Reviews Using the Supervised Machine Learning	553
Sangeeta Bishnoi and Rajendra Purohit	
DiabeDetect: A Novel Decision Tree-Based Approach for Early Prognosis of Diabetes	563
Muhammad Usama Islam, Md. Mobarak Hossain, Iqbal Hossain, and Mohammad Abul Kashem	
Waste Segregator: An Optimized Neural Learning Approach Towards Real-Time Object Classification	573
Drishti Singh, E. Manoj, and T. Anjali	
Analyzing the Impact of Forensic Accounting in the Detection of Financial Fraud: The Mediating Role of Artificial Intelligence	585
Kamakshi Mehta, Prabhat Mittal, Pankaj Kumar Gupta, and J. K. Tandon	
Knowledge Discovery in Geographical Sciences—A Systematic Survey of Various Machine Learning Algorithms for Rainfall Prediction	593
Sheikh Amir Fayaz, Majid Zaman, and Muheet Ahmed Butt	
Hateful Memes, Offensive or Non-offensive!	609
Sujata Khedkar, Priya Karsi, Devansh Ahuja, and Anshul Bahrani	
Drowsiness Detection System Using PPG Sensor's Measured Physiological Parameter	623
Jyoti Tripathi, Satish Chand, Bijender Kumar, Adrija Ghansiyal, and Anshula Nema	
Deep Ensemble Technique for Short-Term Load Forecasting Using Smart Meter Data	635
A. L. Amutha, R. Annie Uthra, J. Preetha Roselyn, and R. Golda Brunet	
Performance Analysis of ISOWC Link Considering Different System Parameters	645
Sanmukh Kaur, Anurupa Lubana, and Anuranjana	
Video Summarization Using SIFT Features and Niblack's Thresholding	655
Amol Shinde, Dipti Jadhav, and Swapnil Shinde	
A Comprehensive Study on Attention-Based NER	665
Tanvir Islam, Sakila Mahbin Zinat, Shamima Sukhi, and M. F. Mridha	
Sentiment Analysis of Multilingual Mixed-Code, Twitter Data Using Machine Learning Approach	683
Sowmya Swamy, Jyoti Kundale, and Dipti Jadhav	
Residual Decoder based U-Net for Semantic Segmentation	699
Shilpa Elsa Abraham and Binsu C. Kovoor	

Ensembled Approach for Text Summarization	709
Minakshi Tomer, Dishant Rathie, and Manoj Kumar	
Statistical Analysis of Impact of COVID-19 Pandemic on States of India	721
Prerna Pandey, Nikki Saraswat, Priyansh Shukla, Kavita Sharma, and Shiv Naresh Shivhare	
A Review on Evolution of Architectures, Services, and Applications in Computing Towards Edge Computing	733
Pranay D. Saraf, Mahip M. Bartere, and Prasad P. Lokulwar	
Image Retrieval Using Multilayer Bi-LSTM	745
Shaily Malik, Poonam Bansal, Pratham Sharma, Rocky Jain, and Ankit Vashisht	
Analysis of FSO-OFDM System Performance for Different Bit Rates and Link Ranges	757
Nabadh Bhan and Sanmukh Kaur	
Customer Churn Prediction in Telecommunication Using Gradient Boosting Machine	769
Manoj Kumar and Dharmendra Kumar Yadav	
Blood-Based DNA Methylation Marker Identification for Parkinson's Disease Prediction	777
Jisha Augustine and A. S. Jereesh	
Disease Detection and Prediction Using the Liver Function Test Data: A Review of Machine Learning Algorithms	785
Ifra Altaf, Muheet Ahmed Butt, and Majid Zaman	
Performance Assessment of Health Decision-Making Using Various Artificial Intelligence Techniques and Evolutionary Algorithms	801
Prabhav Jain, Ekansh Chauhan, and Varun Goel	
Bee Intelligence-Guided Partitional Clustering for Outlier Detection	813
M. Rao Batchanaboyina and Naga Raju Devarakonda	
Author Index	827

About the Editors

Dr. Ashish Khanna has 16 years of expertise in Teaching, Entrepreneurship, and Research & Development. He received his Ph.D. degree from National Institute of Technology, Kurukshetra. He has completed his M. Tech. and B. Tech. GGSIPU, Delhi. He has completed his postdoc from Internet of Things Lab at Inatel, Brazil and University of Valladolid, Spain. He has published around 55 SCI indexed papers in IEEE Transaction, Springer, Elsevier, Wiley and many more reputed Journals with cumulative impact factor of above 100. He has around 120 research articles in top SCI/ Scopus journals, conferences and book chapters. He is co-author of around 30 edited and text books. His research interest includes Distributed Systems, MANET, FANET, VANET, IoT, Machine learning and many more. He is originator of Bhavya Publications and Universal Innovator Lab. Universal Innovator is actively involved in research, innovation, conferences, startup funding events and workshops. He has served the research field as a Keynote Speaker/ Faculty Resource Person/ Session Chair/ Reviewer/ TPC member/ post-doctorate supervision. He is convener and Organizer of ICICC conference series. He is currently working at the Department of Computer Science and Engineering, Maharaja Agrasen Institute of Technology, under GGSIPU, Delhi, India. He is also serving as series editor in Elsevier and De Gruyter publishing houses.

Dr. Deepak Gupta received a B.Tech. degree in 2006 from the Guru Gobind Singh Indraprastha University, India. He received M.E. degree in 2010 from Delhi Technological University, India and Ph. D. degree in 2017 from Dr. APJ Abdul Kalam Technical University, India. He has completed his Post-Doc from Inatel, Brazil. With 13 years of rich expertise in teaching and two years in the industry; he focuses on rational and practical learning. He has contributed massive literature in the fields of Intelligent Data Analysis, BioMedical Engineering, Artificial Intelligence, and Soft Computing. He has served as Editor-in-Chief, Guest Editor, Associate Editor in SCI and various other reputed journals (IEEE, Elsevier, Springer, & Wiley). He has actively been an organizing end of various reputed International conferences. He has authored/edited 50 books with National/International level publishers (IEEE,

Elsevier, Springer, Wiley, Katson). He has published 180 scientific research publications in reputed International Journals and Conferences including 94 SCI Indexed Journals of IEEE, Elsevier, Springer, Wiley and many more.

Prof. Siddhartha Bhattacharyya FIET (UK), is currently the Principal of Rajnagar Mahavidyalaya, Birbhum, India. Prior to this, he was a Professor in Christ University, Bangalore, India. He served as Senior Research Scientist at the Faculty of Electrical Engineering and Computer Science of VSB Technical University of Ostrava, Czech Republic, from October 2018 to April 2019. He also served as the Principal of RCC Institute of Information Technology, Kolkata, India. He is a co-author of 6 books and a co-editor of 75 books and has more than 300 research publications in international journals and conference proceedings to his credit. His research interests include soft computing, pattern recognition, multimedia data processing, hybrid intelligence and quantum computing.

Prof. Aboul Ella Hassanien is the Founder and Head of the Egyptian Scientific Research Group (SRGE) and a Professor of Information Technology at the Faculty of Computer and Artificial Intelligence, Cairo University. Professor Hassanien is an ex-dean of the faculty of computers and information, Beni Suef University. Professor Hassanien has more than 800 scientific research papers published in prestigious international journals and over 40 books covering such diverse topics as data mining, medical images, intelligent systems, social networks, and smart environment. Prof. Hassanien won several awards, including the Best Researcher of the Youth Award of Astronomy and Geophysics of the National Research Institute, Academy of Scientific Research (Egypt, 1990). He was also granted a scientific excellence award in humanities from the University of Kuwait for the 2004 Award and received the scientific - University Award (Cairo University, 2013). Also, He was honored in Egypt as the best researcher at Cairo University in 2013. He was also received the Islamic Educational, Scientific and Cultural Organization (ISESCO) prize on Technology (2014) and received the State Award for excellence in engineering sciences 2015. He was awarded the medal of Sciences and Arts of the first class by the President of the Arab Republic of Egypt, 2017.

Dr. Sameer Anand is currently working as an Assistant professor in the Department of Computer science at Shaheed Sukhdev College of Business Studies, University of Delhi, Delhi. He has received his M.Sc., M.Phil, and Ph.D. (Software Reliability) from Department of Operational Research, University of Delhi. He is a recipient of ‘Best Teacher Award’ (2012) instituted by Directorate of Higher Education, Govt. of NCT, Delhi. The research interest of Dr.Anand includes Operational Research, Software Reliability and Machine Learning. He has completed an Innovation project from the University of Delhi. He has worked in different capacities in International Conferences. Dr. Anand has published several papers in the reputed journals like IEEE Transactions on Reliability, International journal of production research (Taylor & Francis), International Journal of Performability Engineering etc. He is a member

of Society for Reliability Engineering, Quality and Operations Management. Dr. Sameer Anand has more than 16 years of teaching experience.

Dr. Ajay Jaiswal is currently serving as an Assistant Professor in the Department of Computer Science of Shaheed Sukhdev College of Business Studies, University of Delhi, Delhi. He is co-editor of two books/Journals and co-author of dozens of research publications in International Journals and conference proceedings. His research interest includes pattern recognition, image processing, and machine learning. He has completed an interdisciplinary project titled “Financial Inclusion-Issues and Challenges: An Empirical Study” as Co-PI. This project was awarded by the University of Delhi. He obtained his masters from the University of Roorkee (now IIT Roorkee) and Ph.D. from Jawaharlal Nehru University, Delhi. He is a recipient of the best teacher award from the Government of NCT of Delhi. He has more than nineteen years of teaching experience.

Automatic Removal of Eye Blink Artefacts from EEG Data Using Spatio-Temporal Features



Rakesh Ranjan, A. Prabhakara Rao, and Anish Kumar Vishwakarma

Abstract Electroencephalography (EEG) is a standard method in which electrical signals of cerebral activities are collected using electrodes set along the scalp. It is a non-invasive technique used for clinical applications. The common problem with EEG data is that it is susceptible to the imitation of the distinct biological or environmental noise interferences well known as artefacts. Researchers have proposed various methods to eliminate different types of artefacts from the contaminated EEG signal, yet nothing stands standard for endorsement of recorded EEG signals in clinical usage. Consequently, exploration of artefacts removal research remains engaging and challenging even today. In this paper, simulations are performed using a proficient fast independent component analysis (ICA)-based automatic EEG artifact detection based on joint use of spatial and temporal features (ADJUST) method to remove ocular artefacts from EEG signal using the spatio-temporal features. The proposed algorithm has the major function of signal disintegration, temporal and spatial features computation, and classification. This algorithm differentiates artefacts from the independent components segment. The proposed method follows the binary classification of the artefact or non-artefact present in the EEG signal in the pre-final stage while in the subsequent stage, the artefact correction is done. Performance of this algorithm is compared with another well-known ICA algorithm (ARA, i.e. artefact removing algorithm) and the proposed algorithm is producing 18% better outcomes than ARA. Hence, this proposed idea removes the artefacts from the contaminated EEG signal effectively, which might help the neurologists acquire the right information from EEG and enable proper clinical diagnosis.

R. Ranjan (✉)

Department of ECE, National Institute of Technology Patna, Bihar 800005, India

e-mail: rakesh.ec19@nitp.ac.in

A. P. Rao

Department of ECE, Vishnu Institute of Technology, Bhimavaram, AP 534202, India

A. K. Vishwakarma

Department of ECE, VNIT, Nagpur, Maharashtra 440010, India

Keywords EEG signal · Ocular artefact · Eye blinks · Independent component analysis · ADJUST

1 Introduction

Electroencephalogram (EEG) is a common approach for gathering electrical activities of the huge number of neurons within the brain employing EEG signal recording devices [1]. The EEG signal is amazingly stochastic, continuous time-varying, and non-stationary physiological signal [2–4]. EEG is measured using electrodes and conductive material placed at the scalp in accordance with the “10–20” International standard system. The routine activity manifests itself in rhythms having bands of frequency from 0.5 to 60 Hz and 10 to 100 μV range of amplitudes of voltage [5]. Because of temporal resolution and regular declining expenses of EEG recording systems (in contrast with other cerebrum movement recording frameworks like MEG), human brain functionalities are estimated through EEG, a broadly conveyed technique [6]. The most common problem associated with the EEG data is that it is susceptible to interferences due to the distinct biological or atmospheric noise resulting in artefacts. Physiological artefacts like body or eye movements might prompt outrageous amplitudes hops that are multiple times higher than normal brain activity measured with the EEG recording framework [7]. The presence of any artefacts in the EEG causes difficulties for the neurologist to analyse the signal. Ocular artefacts are the most influential artefacts, which are basically originated from eyelid movements, eye blinking or eye fluttering.

In this work, an ICA-based artefact rejection tool—Automatic EEG artifact detection based on joint use of spatial and temporal features (ADJUST) is chosen as a layout to enhance eye blink artefact rejection. As far as the correction of artefacts in biomedical signals is concerned, the necessary background activity should not be altered, leading to improper diagnosis. Independent Component Analysis (ICA) can be described as the statistical technique for deteriorating a bunch of multivariate random vectors into its statistical independent components (ICs) [8]. In this method, the artefacted signals are converted into ICs. The process crumbles EEG signals from multiple channels into spatially fixed and temporally independent components. ICA is a computationally efficient algorithm and shows overall execution even if the acquired data is heavy. Each ICA algorithm is driven by source signals which are expected to be distributed in a non-Gaussian manner while the noise signal is exactly oppositely distributed. ICA method commonly utilizes the Centring, Whitening and decrease in measurements according to the processing task. ADJUST or Automatic EEG artefact detection is a free EEG LAB plug-in tool based on both the spatial and temporal features and is mainly used for artefact elimination or correction present in EEG data [9, 10]. It follows the unsupervised process of multi-dimensional signal decomposition into ICs and classifies the EEG signal into “with-artefact” and

“without-artefact” classes. This tool can handle all kinds of ocular artefacts associated with EEG signals during its recording. This paper is focused on the most influential ocular artefact, i.e. eye blink artefact.

The proposed method shows its potential in correcting the ocular artefacts, and is sustained to hold the necessary background activity. It also enables real-time ocular artefact correction in EEG. Contrasted with ADJUST, the new algorithm includes some additional steps, similar to eye blink artefact recognition and disintegration of the signal into epochs holding artefact and close by clean cerebrum activity samples. Then the extracted epochs are disintegrated with ICA into independent components [9]. The temporal and spatial features of extracted components are calculated and classified into two classes, which are either artefact or non-artefact. A repeating ICA is then executed to estimate spatial-temporal characteristics and classify IC, confirming better artefact rejection results. The present work examines the fast ICA-based ADJUST algorithm’s utility to remove eye blink artefacts present in the EEG signal. Experimental design and EEG data acquisition process are discussed in Sect. 2. The methodology of the proposed artefact elimination algorithm is discussed in Sect. 3. Results are summarized in Sect. 4 and finally, the conclusions are presented in the last section of this paper.

2 Experimental Design and Data Acquisition

Datasets are taken from the Open Vibe repository [11]. The experiment was designed with a typical auditory-event-related potential paradigm for acquiring the EEG data. The audio contained six different words that were played arbitrarily on the speaker. The events were labelled as both targeted as well as non-targeted events. During data acquisition, the subject responds with an eye blink after listing to the targeted words. EEG signal was gathered with two distinctive commercially popular EEG recording frameworks: Emotiv Epoc+ and Brain Products V-Amp [12]. Emotiv Epoc+ is a low-cost and most accessible EEG recording device with 14 channel-saline-based electrodes with a maximum sampling frequency of 128 Hz. On the other side, V-amp is a costly and high-end device for EEG recording. The system has 16 data channels with the maximum sampling frequency of 500 Hz and having 24-bit ADC. Customized electrode configuration is the main advantage of the V-amp device over Emotiv Epoc+.

3 Proposed Enhanced Artefact Elimination Algorithm

Eye blinks are one of the most exceedingly terrible turbulences in EEG signals. These turbulences are portrayed by sudden amplitude enhancement on the frontal electrode of the EEG procurement framework, however, they can prompt amplitude enhancement of other electrodes present in the vicinity. Initially, eye blink was identified

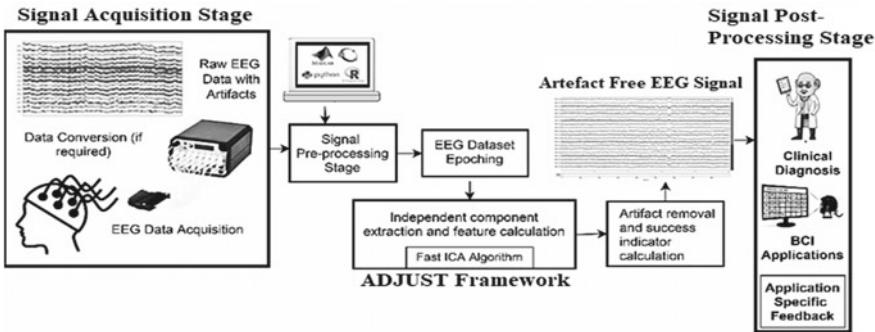


Fig. 1 Block schematic of the proposed eye blinks artefact removal system

by threshold technique as abrupt amplitude jump signified it. The EEG recording framework, reference electrode locations, and peak detection are applied to various electrodes. In the Emotiv EPOC+ framework, four EEG signals from frontal electrodes (AF3, F3, AF4 and F4) are summarized to improve the artefact to clean EEG signal ratio. The eye blink artefact location can be roughly obtained from the peak detection approach. These sample values will be inside spared as markers, as a beginning point for signal epoching and artefact eliminating measure. To obtain better outcomes, the reduction of EEG samples and the calculation of quality measures for each detected eye blink artefact have been the prime focus of this article.

3.1 EEG Data Pre-Processing

Data pre-processing is the beginning stage of the proposed model. After acquiring EEG signals directly from the EEG recording framework or datasets, the pre-processing of these data is done in MATLAB, Python or any other statistical software tools. Initially, the most corrupted part of the signal is simply discarded by the visual inspection method. DC component is also removed from EEG samples in the pre-processing stage only. At the final stage of data pre-processing, the EEG measurements have been filtered using a 16th order FIR band-pass filter in the frequency range from 0.5 to 20 Hz. The signal filtration is done in MATLAB using `filtfilt` function, and it does not alter the phase information. Figure 1 represents the block schematic of the proposed eye blink artefact removal system from the EEG signal.

3.2 EEG Dataset Epoching

We have developed a primary epoching method that divides the complete EEG data into fixed-length epochs. Although the epoch window length can be customized

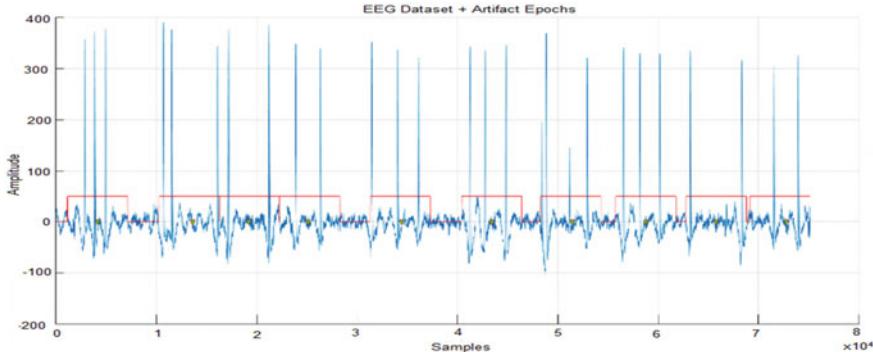


Fig. 2 EEG signal with epochs after applying EEG dataset epoching

manually, we need to set the window length before initializing the process. Shifting epochs from one side to another side is a controlled activity that can prompt the ejection of artefacts from the defined epoch. Following the last limit, epochs moving are conceivable entirely under the condition that all the artefacts encased by epoch in statement remain in a similar epoch after successful shifting. Notwithstanding, it is not generally conceivable to encompass the artefacts with default epoch length without overlapping neighbour epochs. In such cases, the epoch length is diminished by the adjustable number of samples, and the cycle repeats. This action repeats iteratively until the algorithm has figured out how to set the epochs without any overlapping. We do use markers to mark up the region of conceivable eye blinks. The markers are utilized to separate the artefact component parts present in the EEG signal epoch. The iterative independent component analysis is adopted to rectify the artefact parts present in the EEG signal. Figure 2 represents the epoching method output having both EEG signal and epochs.

3.3 *Independent Components Extraction and Artefact Removal Success Indicator Calculation*

ICA algorithm implementation is the next step to be applied on the pre-processed EEG data and epoching of signal for artefact removal. ADJUST uses a fast ICA algorithm for eye blink artefact rejection assignment. Before applying ICA, the signal dimensions are reduced using Principal Component Analysis (PCA) algorithm. The significant parts of ICA are: mixing matrix A, independent components (ICs) and weight matrix W (inverse of mixing matrix A). The determination of spatial and temporal features depends directly on ICA composite matrices. Apart from the above-said three parts of ICA, the fourth component is the convergence success indicator. The unsuccessful convergence indicator shows that incorrect results are obtained using a given ICA. The ICA needs to perform iteratively until the convergence

indicator is successful or the iteration counter achieves the maximum permissible values. After every iteration, new initial values of matrix A regenerate. In the case of the maximal permissible iteration of ICA, the ICs in the epoch cannot be isolated. Hence, influenced epoch will be marked and the posterior artefact removal process will be terminated.

Steps for Fast ICA algorithm

- i. $Y \leftarrow$ input (mixed-signal)
 - ii. $Y \leftarrow Y - \text{mean}(Y)$ // Finding the center of the data
 - iii. $[E D] \leftarrow \text{eigen}(Y^T Y)$
 - iv. $Z \leftarrow Y * D^{-1/2} * E^T$ // Whitening the data
 - v. $W \leftarrow \text{rand}(\text{size of}(Y))$; // Matrix of Random numbers
 - vi. $W \leftarrow E \{ Y (W^T Y)^{-1} \} - 3W$
 - vii. $W \leftarrow W - BB^T W$ // B is the matrix whose columns are found.
 - viii. $W = W / \|W\|$
 - ix. if $|W^*W+|=1$ goto “x”, else goto “vi”,
 - x. $Y \leftarrow W^T Y$ //Solution
 - xi. $\text{Output}() \leftarrow Y$
-

3.4 Artefact Removal and Success Indicator Calculation

The spatial and temporal features are derived indirectly from the mixing matrix and ICs of the ICA algorithm. The median of feature values is calculated, and according to median values, the EEG signal is classified into artefact-free signal and artefacted signal. The two classes individually form two different Gaussian distributions. The intersections of these two developed Gaussian curves have been the threshold value for this artefact classification task. The threshold comparison is done between spatio-temporal feature value of every ICs and obtained corresponding threshold value, thus classification task has been accomplished. It can be observed that ICs on the left side of the threshold are fallen into ICs with no artefacts while the right of the threshold components are artefacted ICs. Those artefacted ICs have been eliminated from EEG signal by applying an iterative ICA algorithm. The above procedure is repeated for all the available epochs. The final de-noised version will be the concatenation of the samples of all de-noised epochs. Artefact containing sharp edges in the processed EEG signal are removed by applying filters.

After eye blink artefact elimination from EEG signal, artefact removal success indicator needs to be determined for every epoch. The unsuccessful indicator suggests that the artefact is not correctly identified. In these circumstances, the algorithm is performed repeatedly until valid results are obtained. Whenever the iteration repeats, a new initial value of matrix A is generated, and statistical parameters (mean and standard deviation) are updated. Standard deviation determination plays a vital role

in comparing the success estimation of any EEG artefact removal algorithm. If any unknown signal or epoch's standard deviation is slightly more than the standard deviation of the cleaned EEG signal, the signal epoch is marked as artefacted signal. In the case of reaching of maximal permissible iteration of ICA, isolation of the ICs in epoch is not achieved; the posterior artefacts removal process will be terminated after marking the influenced epoch. Figure 3 represents a clean EEG signal and artefacts present in the EEG signal. Figure 4 illustrates the EEG signal after removing artefact from ADJUST and ARA algorithms, and Fig. 5 describes the flow of the proposed work.

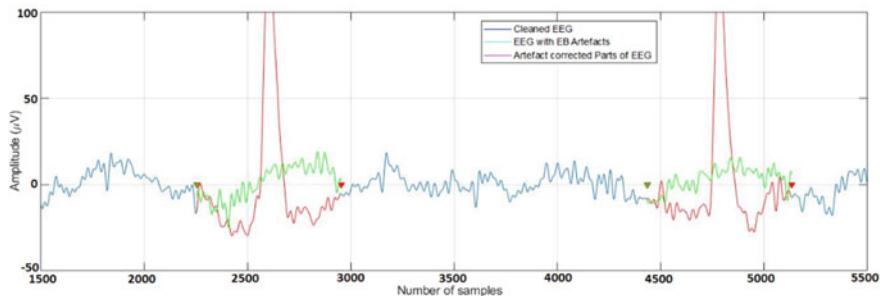


Fig. 3 EEG signal showing artefact epoch with two artefacts (Left and right side of the artefact are marked with green and red triangles, respectively). Clean/normal EEG signals are shown in blue lines while the eye blink artefacts are represented in red lines. Signals in green are the corrected parts of EB artefacts present in the EEG signal

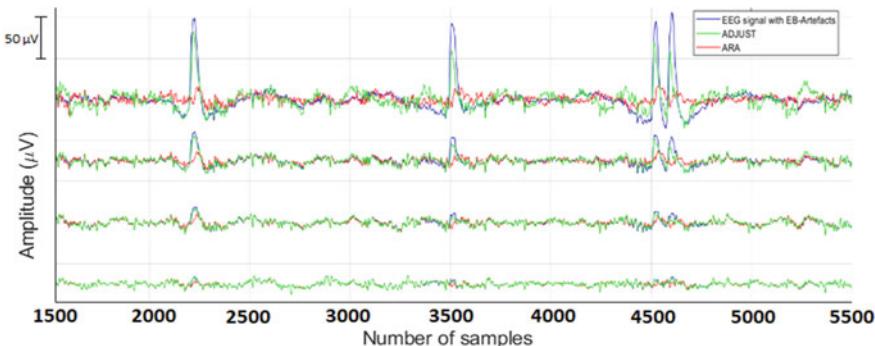


Fig. 4 A portion of the EEG signal after removing artefacts using ADJUST (green lines) and artefact removal algorithm (ARA) (red lines). Original EEG signals with artefacts are shown in blue lines. The EEG recording output shown here are mainly taken from four prominent electrode positions, i.e. Fp1, Fz, Cz and Pz

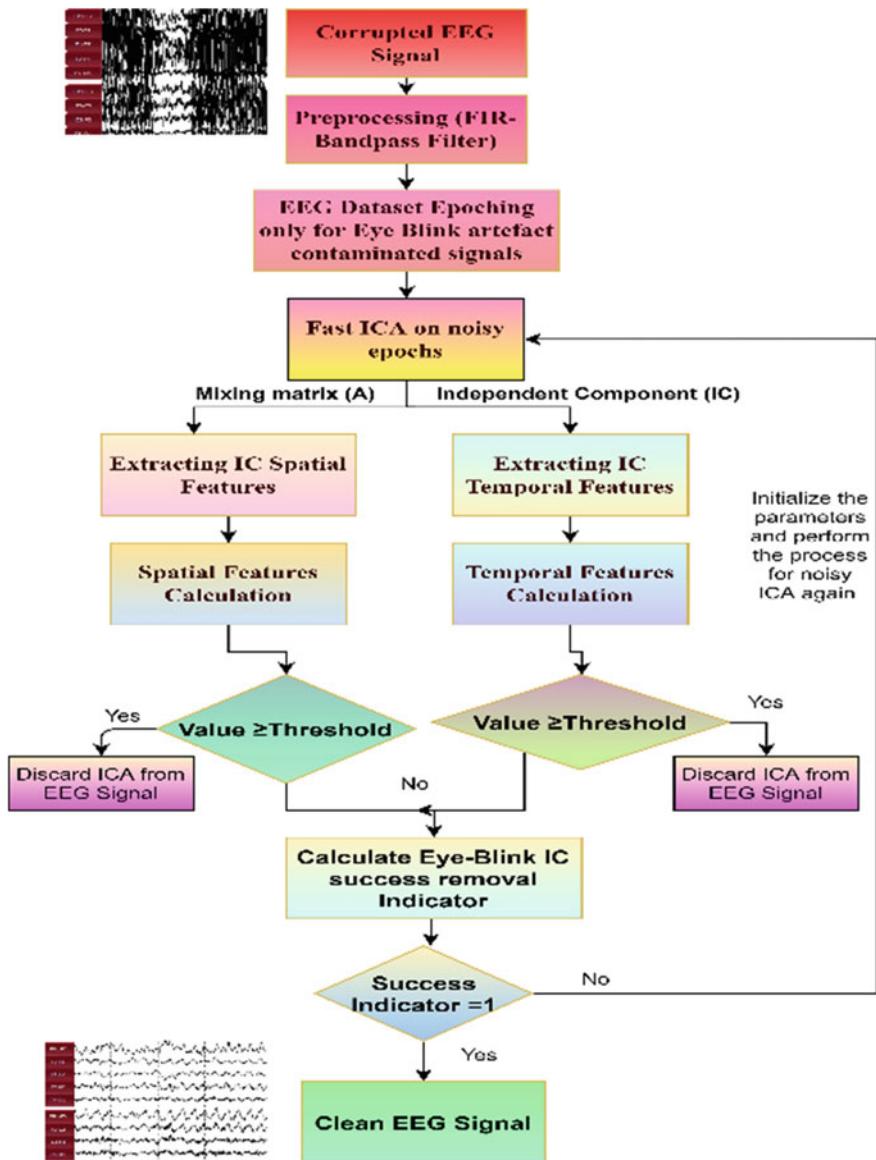


Fig. 5 Flow chart of the proposed work

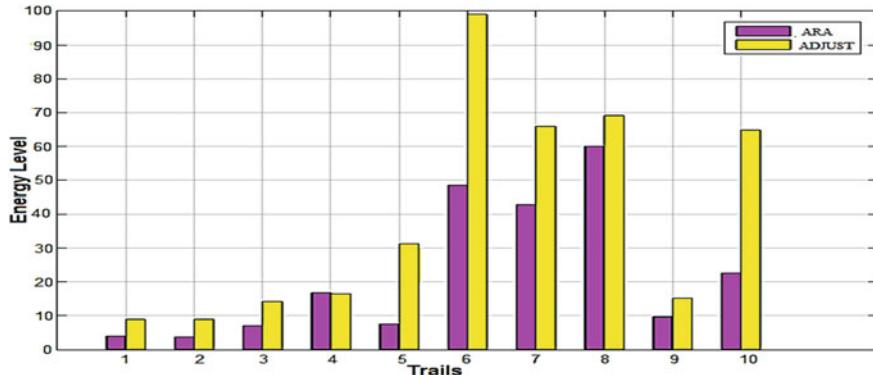


Fig. 6 Energy level variation in EEG signal after the successful removal of eye blink artefacts using ADJUST and ARA over ten successive trails

4 Results

Experimental trials were directed on 12 channels of the EEG recording system. We have conducted a simulation task for eye blink artefact elimination using ICA based ADJUST algorithm and achieved good results. The process is validated with another popular algorithm mostly applied for EEG signal artefact correction, which is Artefact Removal Algorithm (ARA). The waveform results have been depicted in the preceding figures. The EEG recording output is mainly taken from four prominent electrode positions, which are prefrontal (Fp1), frontal (Fz), central (Cz) and central parietal lobe (Pz). Figure 6 represents the energy-level variation in EEG signal after the successful removal of eye blink artefacts over ten successive iterations. We have observed that the results improved around 18% after ten subsequent iterations using ADJUST method for eye blink artefacts removal.

5 Conclusion

Brain activity and its signal processing play a vital role in neurology, neuroscience and neural engineering. EEG signals are critically important in recognizing numerous psychological problems and BCI applications. In this paper, an efficient ICA algorithm (fast ICA-based ADJUST) is utilized to isolate the EEG signal from the contaminated EEG signal. In this process, the corrupted epochs are decomposed into independent components using the ICA algorithm. Repetitive ICA executions with feature threshold comparison in spatial and temporal domain simultaneously ensure better results. The implemented algorithm is validated with another well-known ICA algorithm (ARA, i.e. artefact removing algorithm), and 18% better outcomes are achieved over ten trials iteratively.

References

1. J.C. Henry, Electroencephalography: basic principles, clinical applications, and related fields. *Neurology* **67**(11), 2092 (2006)
2. I. Daly et al., What does clean EEG look like? in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (2012), pp. 3963–3966
3. R. Ranjan, R. Arya, P. Kshirsagar, V. Jain, D.K. Jain, A.K. Sangaiah, Real time eye blink extraction circuit design from EEG signal for ALS patients. *J. Med. Biol. Eng.* **38**(6), 933–942 (2018)
4. A. Bisht, C. Kaur, and P. Singh, “Recent advances in artifact removal techniques for EEG signal processing, in *Intelligent Communication, Control and Devices* (Springer, 2020), pp. 385–392
5. S.A. Zamin, M. Awais, B. Altaf, W. Saadeh, A single channel EEG-based all AASM Sleep stages classifier for neurodegenerative disorder, in *2019 IEEE Biomedical Circuits and Systems Conference (BiCAS)* (2019), pp. 1–4
6. D.P. Subha, P.K. Joseph, R. Acharya U, C.M. Lim, EEG signal analysis: a survey. *J. Med. Syst.* **34**(2), 195212 (2010). doi: <https://doi.org/10.1007/s10916-008-9231-z>
7. N. Noury, J.F. Hipp, M. Siegel, Physiological processes non-linearly affect electrophysiological recordings during transcranial electric stimulation. *Neuroimage* **140**, 99–109 (2016)
8. J. Iriarte et al., Independent component analysis as a tool to eliminate artifacts in EEG: a quantitative study. *J. Clin. Neurophysiol.* **20**(4), 249–257 (2003)
9. A. Delorme, S. Makeig, EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neurosci. Methods* **134**(1), 9–21 (2004)
10. A. Mognon, J. Jovicich, L. Bruzzone, M. Buiatti, ADJUST: an automatic EEG artifact detector based on the joint use of spatial and temporal features: automatic spatio-temporal EEG artifact detection. *Psychophysiology* **48**(2), 229–240 (Feb. 2011)
11. OpenViBE | Software for Brain Computer Interfaces and Real Time Neurosciences. <http://openvibe.inria.fr/>
12. A. Puce, M.S. Hämäläinen, A review of issues related to data acquisition and analysis in EEG/MEG studies. *Brain Sci.* **7**(6), 58 (2017)

Autoencoder-Based Model for Detecting Accounting Statement Fraud



Hiral A. Patel, Abhishek Parikh, and Bhargav Patel

Abstract The past two decades have witnessed several captions with incidents of corporate financial frauds in India. There is huge destruction in the wealth of investors as well as the society due to financial frauds. Not only local but also direct foreign investors may also give a second thought before investing in Indian companies. As per the cases registered in the recent past, especially Satyam and Wipro-World Bank have made investors rethink about investing in Indian companies with regard to the Indian business framework. In parity with developing countries, there are several cases of fraud in developed countries too. These cases created the urgency to develop the mechanism of detecting financial fraud worldwide in advance. If the fraud can be detected at an early stage, then investors can protect themselves from financial losses. These cases emphasize the need for all stakeholders to protect their investments by discovering fraud in its primitive stages by distinguishing between truthful and false information. For the current study, the researcher has collected 1000 data points of Non-Fraudulent companies and 86 data points of Fraudulent companies for the period of 2008–2020 for Indian manufacturing companies listed on the Bombay Stock Exchange. Artificial Neural Network Autoencoder model has been trained to learn the representation of the data of Non-fraud companies. This learning has been used to classify the financial statements among fraud and non-fraud companies. To train the model, 800 data points of only non-fraudulent companies were used. However, 200 data points of non-fraudulent with 86 data points of fraudulent companies were used for testing the model. The researcher is able to achieve 98.95% accuracy in the classification of fraudulent as well as non-fraudulent companies.

H. A. Patel (✉) · A. Parikh
Ganpat University, Ganpat Vidyanagar, Mehsana 384012, India
e-mail: hiral.patel@ganpatuniversity.ac.in

A. Parikh
e-mail: abhishek.parikh@ganpatuniversity.ac.in

B. Patel
School of Engineering and Applied Sciences, Ahmedabad University, Ahmedabad, India

Keywords Financial fraud detection · Model development · Artificial neural network · Autoencoder

1 Introduction

The roots of the corporate world are quivering because of the financial disaster by leading companies like Enron, Lucent, WorldCom and UTI [22]. Although the corporate world had not retrieved completely from such distortion, companies like Satyam Computers in India joined the cluster of companies who committed frauds in financial statements. Along with Satyam Computers, AIG and Lehman Brothers enrolled as the companies with financial frauds. Such financial fraud implosions made it difficult for financial investors to trust the performance of the corporate world. Specifically, in a country like India, where the government is trying hard to improve the economic growth of the country, it is equally important to detect fraudulent activities done by listed companies beforehand.

In the twenty-first century, the scandals related to financial frauds have triggered the necessity of improving the skills of auditors to perceive in it the early phase. As indicated in previous research, financial scandals and frauds are the main reason for the economic and financial crisis. It has a direct linkage with the performance of capital, equity, and debt market [9]. It created an urgent requirement to take safety measures to improve the efficiency and liquidity aspects. To reduce the turbulence and uncertainty in the economy, the rights of investors must be safeguarded. At the same time, to prevent the investors from huge losses, it is imperative for technology specialists and financial analysts to study it very well. The action needs to improve the confidence of investors worldwide. It will also significantly contribute to strengthen the credibility of information shared by the company and increase the conviction of investors in their overall financial investment decisions [19].

Association of Certified Fraud Examiners (ACFE) [1] explains fraud is “deception or misrepresentation made by an individual or entity, knowing that the misrepresentation could result in some unlawful benefit to the entity or to the individual or to some other party” [1]. Market participants such as investors and creditors are making the decision of investing in any company based on the financial information in the form of financial statements published/available. The reliability, uniformity, and transparency of the published information is critical for an investor to take a wise decision of investing. Therefore, it is the reverent job and responsibility of the auditors, society, business fraternity, and different regulators to thwart financial statement fraud. Undoubtedly, its occurrences create a confidence crisis in the corporate world and ultimately affect the economy of a country [23].

Traditionally, auditors are responsible for spotting financial statement fraud [7]. Detecting fraud through normal audit measures is a difficult task [18] because with the appearance of an increasing number of companies that exercise these unfair practices, overburdened the auditors with the task of detection of fraud. Also, there is an acute shortage of auditors with knowledge concerning the characteristics of financial fraud.

Moreover, most auditors lack the know-how and experience necessary to detect it because of its uncommonness and irregularity of fraudulent activities [8]. This limitation suggests it is imperative to detect false financial statements for additional analytical procedures. In this research, the autoencoder model was developed and trained for classifying fraud and non-fraud financial statements. Autoencoders are a kind of neural network that is used to powerfully learn the data representation or representation space in an unsupervised manner. The main objective is to learn a reduced representation of the input data.

2 Literature Review

Agbaje Wale Henry and Dare Funso (2018) [13] specified the objective focusing on establishing the consequence of variables of Financial Statement Fraud on ROA (Return on Assets). For accomplishing their goal, they adopted a descriptive research design. For their purpose of study, they collected secondary data from the Financial Reports of the chosen companies. For analysis of data, they used ANCOVA Analysis of Covariance) and an economic method STATA II. They adopted the Beneish Model for the analysis of financial reports to create a dummy variable for the chosen firms. For validation of the parameters, they used 4 statistical techniques, namely Wald chi-square, co-efficient of determination, t-test, and F-statistics. After that, they made a Regression model to find the relation of Return on Assets with fictitious revenue by making the hypothesis. They concluded that fictitious revenue showed a negative relationship with the firms' profitability, this means that when the firm revenue is declared or recognized to be fictitious, the Return On Asset (ROA) will decrease 5.07%. Marsellisa Nindito (2018) [16] researcher made a pentagon model to find out the effect of 5 parameters namely rationalization, pressure, arrogance, capability, and opportunity. For making this model he selected 14 companies from the financial services sector which were listed on the Indonesia Stock Exchange during 2013–2015 and he selected and matched other 14 companies that were similar both from the aspects of industry and size. He did a Logistic Regression analysis to check ten hypotheses on the impact of rationalization, pressure, arrogance, capability, and opportunity on possible fraud reporting. It was found in the result that pressure, opportunity, and capability have a significant impact on FFS.

Ata, H. A., & Seyrek, I. H. [3] made 2 data mining models, namely Decision tree and Neural network on 100 manufacturing firms listed in Istanbul Stock Exchange (ISE). 24 financial ratios were used as a set of variables. The decision tree model gave 67.92% accuracy and Neural Network gave 77.36% accuracy. Gupta, R., & Gill, N. S. (2012) [12] generated association rules. They considered 62 variables containing financial ratios and financial variables. Their data set contained 114 companies listed in different stock exchanges globally during the year 2007–2011. To reduce the dimensionality, they further applied ANOVA and found 35 informative variables. To derive association rules, 3 data mining techniques, namely Decision Tree, Naïve Bayesian Classifier, Genetic programming were used. The two-performance

matrix sensitivity and specificity had been used. The decision tree produced the best sensitivity and genetic programming produced the best specificity.

Song et al. [20] made 5 classifiers, namely LR, BPNN, C5.0, SVM, and voting ensemble. They considered variables from three categories: Attitude, motivation, and condition. They found that among all these five algorithms, the ensemble algorithm outperformed.

Kotsiantis, Koumanakos, Tzelepis, and Tampakas (2006) [15] considered 164 Greek Manufacturing companies listed on ASE (Athens Stock Exchange). They applied 7 data mining algorithms upon 164 data by using 8 relevant variables. They found that the C4.5 algorithm which falls under the Decision Tree Induction gave the highest accuracy which correctly classifies 81.2% of the total samples. Bell, T. B., & Carcello, J. V. (2000) [5] made a logistic regression model by considering the 77 firms having fraud engagements and 305 firms not having fraud engagements to check the probability of occurring fraud in financial statements. They found that the logistic model was more precise than auditors in checking the risk of fraud for 77 fraud cases. Noteworthy difference was not found among model prediction and auditor's note for non-fraud samples. Patel, H. et al. (2019) [17] collected 86 fraudulent and 92 non-fraudulent transactions of Indian manufacturing companies which listed on the Bombay Stock Exchange during 2008–2017. They considered 31 variables from which they found 10 significant variables on the basis of T-test. They applied 42 data mining algorithms to the collected data set with 10 significant features. Their experiment showed that Random Forest which is an ensemble technique gave the highest result of 89% accuracy of the Model with the help of hyperparameter tuning of Random Forest Classifier.

Prior research in this area used supervised machine learning algorithms to develop the model for detecting financial statement fraud [4–6]. Supervised learning algorithms have been dominant methods for detecting financial statement fraud. Further investigation, focusing on new techniques based on unsupervised learning methods may be beneficial [22]. This research paper explores the possibility of detecting financial statement fraud with the help of an unsupervised learning algorithm. For the current experiment, the researcher collected 1000 non-fraudulent transactions and 86 fraudulent transactions with 10 significant variables and made an autoencoder model to classify fraudulent and non-fraudulent transactions. The detailed study has been thoroughly described in the next section named Research Methodology.

3 Research Methodology

Figure 1 describes the process researchers have utilized for this work. First, data of fraud and non-fraud companies is collected. After collection, data is normalized and labels are removed. The training set contains data of only non-fraud companies. Our objective is to learn the representation of the non-fraud data. To do that, researchers have used the Artificial Neural Network (ANN)-based state-of-the-art deep learning architecture known as autoencoder [10].

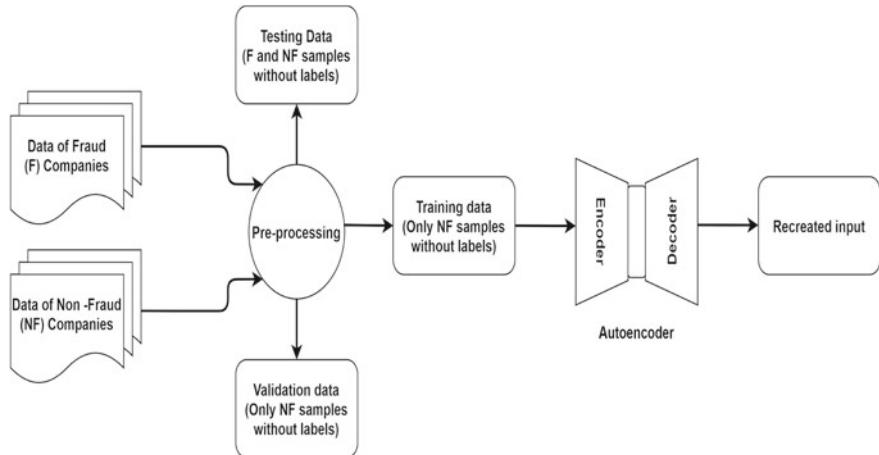


Fig. 1 Process for the model development and testing

3.1 Data and Variables

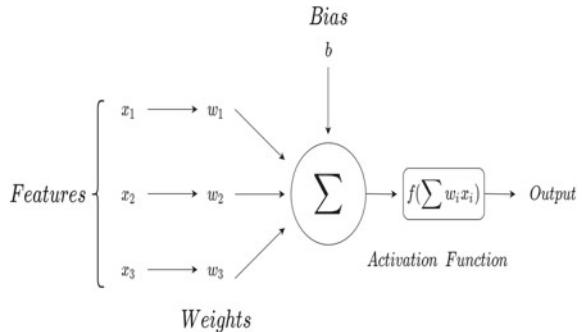
Researchers collected 1000 data points for non-fraudulent companies and 86 data points for fraudulent companies which are listed on BSE during the years 2008–2020. To collect the data, researchers used Capitaline database. The researcher evaluated the auditor's note from the Annual report of the individual company and on the basis of the auditor's remark the company was categorized as either fraudulent or non-fraudulent company. For the current study, the researcher considered 10 important financial ratios as independent variables suggested by Patel, H. et al. (2019) [17], namely Return of asset, Total asset turnover ratio, Earning per share, Equity ratio, Book value per share, Account payable turnover, Return of stake holders' equity, Net working capital, Days inventory outstanding, and Days payable outstanding.

4 Data Analysis and Interpretation

4.1 Artificial Neural Network

ANN is a group of connected nodes or units known as artificial neurons which loosely model the biological neurons in the brain [21]. These neurons work collaboratively to carry out a given task. Each neuron receives an input in the form of numbers. Given input is processed by summing up the weighted inputs and passing it from the nonlinear function known as the activation function. The connection between neurons is known as edges. These edges have weights that are adjusted during the training process. Figure 2 represents the overall process going in the neuron.

Fig. 2 Model process in ANN



4.2 Autoencoder

A simple autoencoder is a type of deep learning architecture that is used to learn the latent representation of the input data in an unsupervised manner. The aim of the autoencoder is to learn a compressed representation of the data in lower dimensions while ignoring the noise in the data. Autoencoders are designed in a way so that a bottleneck is created at the center part. This bottleneck encodes the data into a lower dimension and learns the latent representation of the data. Along with reducing data into the lower dimension, the data reconstruction method is also learned. The encoder encodes the data and the decoder tries to reconstruct the same data. By doing this, the autoencoder learns the latent representation of the data. This learning can be used to detect the abnormality in the data.

Figure 3 shows the autoencoder architecture used in this work. In the figure, we can observe that data is first encoded into a lower dimension by the encoder and then reconstructed using the decoder.

4.3 Network Training

An unsupervised approach is used to train the model. The training procedure is implemented in Python-3 with Keras API [11] and TensorFlow [2] as backend. The Adam [14] optimization algorithm is considered for network training.

As Adam is an adaptive learning rate algorithm, it computes the individual learning rate for different parameters. Mean Absolute Error (MAE) is used to measure training loss. Other parameters are mentioned in Table 1.

Figure 4 shows the comparison of training loss and validation loss. From the figure, it is clear that the model is finally trained. It can be observed that validation loss and training loss goes hand in hand which indicates that the model is generalized well over the training and validation data.

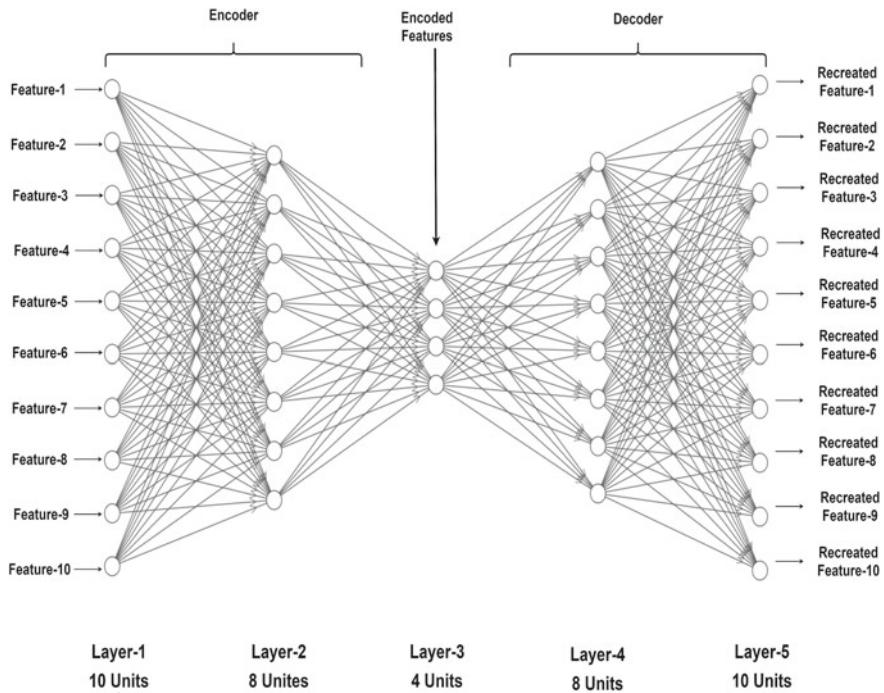


Fig. 3 Autoencoder architecture

Table 1 Value of hyperparameters

Hyperparameters	Value
Initial learning rate	0.001
Batch size	32
Max epochs	200
Number of hidden layers	3
Optimization algorithm	Adam
Activation function	ReLU
Loss function	Mean Absolute Error (MAE)

5 Fraud Detection

Fraud detection is the task of detecting outlier samples. The outlier sample does not follow the distribution of non-fraud data. Firstly, the model is trained on the non-fraud data to learn the latent representation of the true data. Then the threshold is decided on the basis of the distribution of MAE obtained from making predictions on training data and then calculating MAE by comparing with ground truth values.

In Fig. 5, we can observe that the MAE distributions of non-fraud and fraud

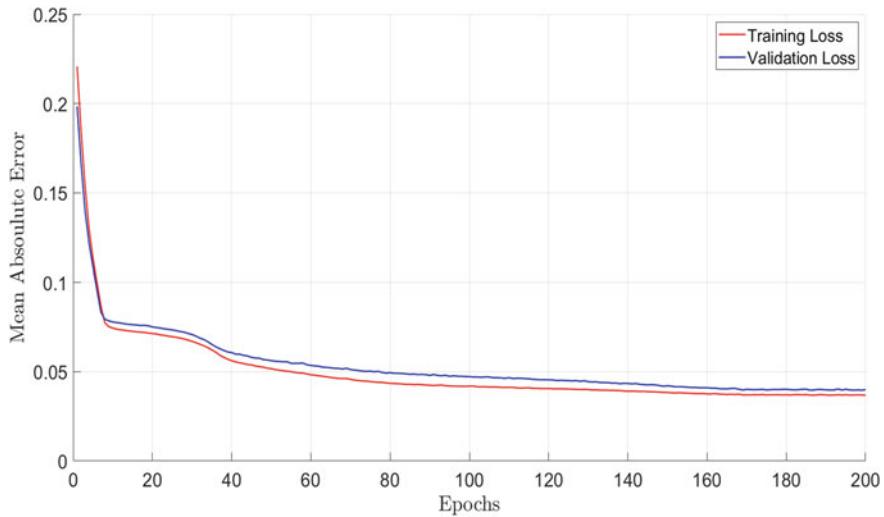


Fig. 4 Model training and validation loss

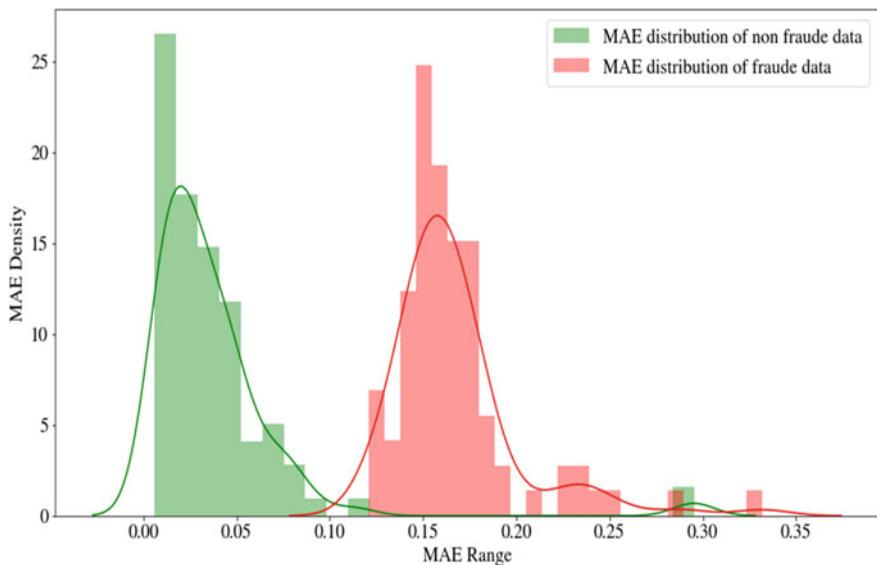


Fig. 5 Mean absolute error

data have different ranges. Researchers have exploited this property and use this to classify the samples. For the scope of this work, we have considered the threshold MAE value as 0.125. To detect the fraud companies in the data, a mix of fraud and non-fraud companies' data is given as an input to the model, as the model has

learned the representation of non-fraud company's data, it will try to map the fraud data to non-fraud data. The output of the model is compared with the input data and MAE is calculated. After this, the MAE of the input data sample is compared with a threshold and the sample which has MAE higher than the threshold is labeled as a fraud company.

6 Discussion

Firstly, we have trained the autoencoder model to learn the latent representation of the non-fraud data. After that, as discussed in Sect. 3-d, the threshold is decided. To test the model, researchers have prepared a test data set, which consisted of 86 fraud data samples and 200 non-fraud data samples.

Figure 6 shows the MAE distribution of fraud and non-fraud samples. It can be observed that the MAE distribution of test data follows the bimodal distribution. The first peak follows the range of non-fraud data and the second peak follows the range of fraud data. This MAE of test data is compared with the threshold and samples are classified. We have been able to achieve 98.95% accuracy in the classification of fraud and non-fraud samples.

Figure 7 shows detailed performance metrics.

Apart from these performance metrics, researchers have also considered other metrics such as precision, recall, and F1-scorer. These performance metrics are given in Table 2.

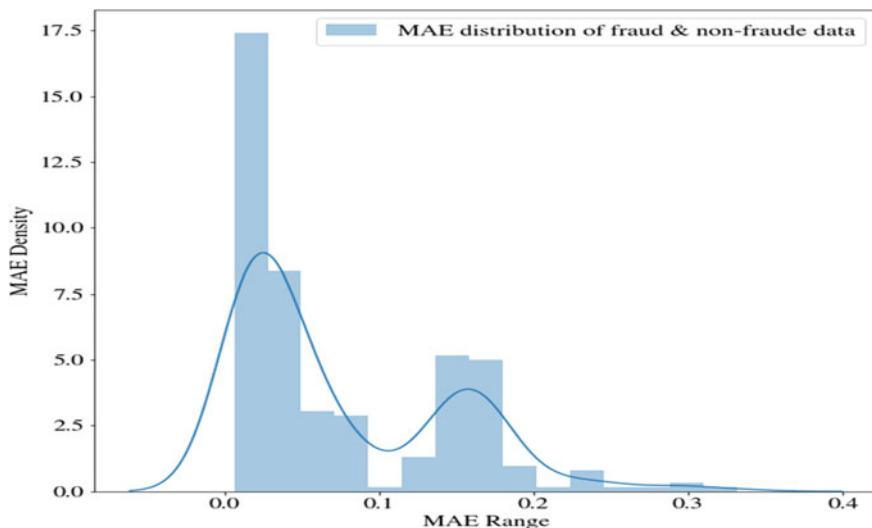
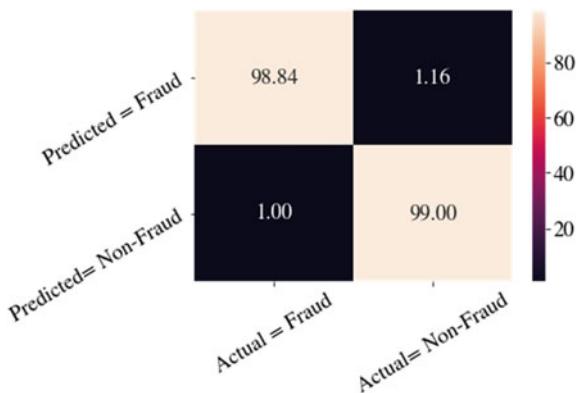


Fig. 6 MAE distribution

Fig. 7 Performance metrics**Table 2** Performance value

Performance metrics	Value (%)
Precision	98.84
Recall	99.0
F1-score	98.82

7 Conclusion

The researcher is successfully able to train the model that helps the investor community to identify possible fraud cases with 98.84 percent accuracy. In addition, it clearly represents the use of Artificial Neural Network for accurate identification of any future wealth destruction. Of course, when the company is fraud and identified as non-fraud it is more dangerous than a company identified as non-fraud when actually it is fraud. Hence, out of 100 predicted non-fraud companies even 1 actually fraud company is open, the scope of working is stronger criteria to develop a better model. This piece of research is giving a new way of prediction for fraud companies using machine learning tools.

References

1. Association of Certified Fraud Examiner, “What is Fraud”, available on <http://www.acfe.com/fraud-101.aspx>
2. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, , J Dean, M. Kudlur., Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 265283 2016
3. H.A. Ata, I.H. Seyrek, The use of data mining techniques in detecting fraudulent financial statements: An application on manufacturing firms. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 14(2) (2009)

4. M.S. Beasley, An empirical analysis of the relation between the board of director composition and financial statement fraud. *Accounting review*, 443465 (1996)
5. T.B. Bell, J.V Carcello, A decision aid for assessing the likelihood of fraudulent financial reporting. *Auditing: A Journal of Practice & Theory*, **19**(1), 169184 (2000)
6. R.A. Bernardi, Fraud detection: The effect of client integrity and competence and auditor cognitive style. *Auditing* **13**, 68 (1994)
7. C.P. Cullinan, S.G. Sutton, Defrauding the public interest: A critical examination of reengineered audit processes and the likelihood of detecting fraud. *Crit. Perspect. Account.* **13**(3), 297–310 (2002)
8. K.M. Fanning, K.O. Cogger, Neural network detection of management fraud using published financial data. *Intelligent Systems in Accounting, Finance & Management* **7**(1), 21–41 (1998)
9. G. Apparao et al., Financial Statement Fraud Detection by Data Mining. *Int. Journal of Advanced Networking and Applications* **159**(1), 159–163 (2009)
10. I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, *Deep learning*, vol. 1, No. 2. (Cambridge: MIT press. 2016)
11. A. Gulati, S. Pal, *Deep learning with Keras*, (Packt Publishing Ltd, 2017).
12. R. Gupta, N.S. Gill, Prevention and detection of financial statement fraud—An implementation of data mining framework. *Editorial Preface* **3**(8), 150–160 (2012)
13. A.W. Henry, D. Funso, Dynamic Analysis of Financial Statement Fraud on Profitability of Manufacturing Firms in Nigeria.
14. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. (2014)
15. S. Kotsiantis, E. Koumanakos, D. Tzelepis, V. Tampakas, Forecasting fraudulent financial statements using data mining. *Int. J. Comput. Intell.* **3**(2), 104–110 (2006)
16. M. Nindito, Financial statement fraud: perspective of the pentagon fraud model in Indonesia. *Academy of Accounting and Financial Studies Journal*. **22**(2), (2018)
17. H. Patel, et al., An application of ensemble random forest classifier for detecting financial statement manipulation of Indian listed companies. In *Recent Developments in Machine Learning and Data Analytics* . (Springer, Singapore, 2019), pp. 349360
18. B. Porter, A. Cameron, Company fraud—what price the auditor?. *Accountant's Journal*, (December), 4447 (1987)
19. Z. Rezaee, B.L. Kedia, Role of Corporate Governance Participants in Preventing and Detecting Financial Statement Fraud. *J. Forensic Investig. Account.* **4**(2), 176–205 (2012)
20. X.P. Song, Z.H. Hu, J.G. Du, Z.H. Sheng, Application of machine learning methods to risk assessment of financial statement fraud: evidence from China. *J. Forecast.* **33**(8), 611–626 (2014)
21. S.C. Wang, Artificial neural network. In *Interdisciplinary computing in java programming* (Springer, Boston, MA, 2003), pp. 81100
22. D. Yue, X. Wu, Y. Wang, Y. Li, C.H. Chu, (September). A review of data mining-based financial fraud detection research, in *Proceedings of the 2007 International Conference on Wireless Communications, Networking and Mobile Computing*, IEEE, (2007), pp. 55195522
23. W. Zhou, G. Kapoor, Detecting evolutionary financial statement fraud. *Decis. Support Syst.* **50**(3), 570–575 (2011)

Increase in Mental Health Cases Post COVID Outbreak



Agnideepa Majumder, Mehardeep Singh Arora, Palak Mantri,
and Ankur Saxena

Abstract The mental health or well-being of an individual is described as his/her state of mind which conjointly provides an outline of that individual's nature. It is primarily the combination of psychological, emotionality, and well-being of an individual socially. The ability of a person to think, feel, and handle situations determines his mental health. An ample of factors result in prior mental illness, for example, stress, depression, anxiety with simultaneous obsessive-compulsive disorder and moreover personality disorders. Right now, we are all facing emotions, thoughts, and situations that we have never been through. In India, the COVID-19 pandemic scenario is having a huge and significant effect on public mental health with regards to their sex, age, profession, socio-economic status, their residing place, etc. The front-line workers are more distressed than the other professionals; the plight of migrants is disturbing; unemployment of huge numbers of people, students, and teachers facing distress as some are unable to afford online platforms and smooth transition to online learning. Therefore, monitoring the mental health of the population during this critical period is an immediate priority. Machine learning algorithm and the pure nature of artificial intelligence (AI) can be used to predict the onset of mental illness. AI is a revolutionary and wide-ranging field of computer science that is involved with performing several tasks that substitute human intelligence by building smart and computational tools and machines. Over the coming years and decades, it has set to become a core component of all modern software. Machine learning is a subset of AI. This research work has employed the application of various machine learning algorithms on the Jupyter platform, such as the k-nearest neighbors (KNN) algorithm and seaborn to determine the state of mental illness in particular target groups. Using these above-mentioned tools, we have generated few graphs that show the stress and depression counts among different age groups. Analyzing the results so obtained in this research paper, we can clearly figure out the appropriate measures that can be taken into consideration for any such dilemma in the near future.

A. Majumder · M. S. Arora · P. Mantri · A. Saxena (✉)
Amity Institute of Biotechnology, Amity University, Noida, Uttar Pradesh, India
e-mail: asaxena1@amity.edu

Keywords Artificial intelligence · Machine learning · Mental health · COVID-19 · KNN · Jupyter · Seaborn

1 Introduction

Mental health is a combination of an individual's emotional, psychological, and social well-being. It affects how a person thinks, feels, or even acts and reacts. It helps to determine how an individual handles the daily stress and relates to others and makes daily life decisions and choices [1, 22].

Mental health is essential at every stage of life, from the day a child is born and adolescence which further finally leads to adulthood. It can even be summed up as the imbalance in brain chemistry. Over the duration of life, a person's thinking, mood, and behavior could be affected. Many factors contribute to the mental health problems which may include:

- Biological factors, such as genes or brain chemistry.
- Life experiences, such as trauma or abuse.
- Family history of mental health problems.

Mental health problems are very common worldwide inclusive of mood swings; personality; lack of sleep; inability to overcome day-to-day activities or stress; and withdrawal from society, family, friends, and certain activities [2, 24]. Coping with the state of mental illness can be physically or we say economically and even emotionally demanding not only to an individual but also to even the people surrounding the patient who is affected. Work disability is one and the foremost key and adverse consequence of the mental disorder. An individual can perceive the change in emotions and even if someone is intoxicated with the mental illness, then also he/she may suffer from the insights and drowns in his/her mental state. The basis of our "mind reading" ability is rooted in the influence (conscious and subconscious) that our mental state has on the cognitive and motor processes that completely control the coordinated production of rich, complex behaviors.

Now the need to organize the mental health profiles of the different participants that belong to different communities in specific scenarios and in order to predict any of the kind of health-related anomalies and disabilities. The community can be considered broadly and majorly classified as high school students, college youths, and working professionals. The effect of the COVID-19 on mental illness of generalized public in India and all over the world has both pros and cons [3, 23].

Growth of fear and anxiety are possibly the two major and most commonly acknowledged emotional responses that any of the public will feel. Gradually, people who bear the lockdown approached toward the release of lockdown. Finding a suitable and subtle way to pull the people who actually suffered through the lockdown took a lot of new emotional energy and people actually got confined and stuck toward becoming habitual with certain places to cope up.

Many people feared becoming ill with the virus or passing the infection on to their loved ones by becoming the carrier of COVID-19 as the risk fluctuated when people begin to interact with each other or even closely associated among each other [4, 21]. People have developed this phobia and they feel nervous or anxious whenever they go back to home from out whether it be the nearest grocery store. Coping with such a new normal lifestyle may have been very ill for people of all ages.

Machine learning concepts have been incorporated and applied to actually study the survey of people that systematically had to deal with the ideology of lockdown and then impregnating to predict the user-friendly contextual information which broadly signifies and highlights such as the mood of the individual, physical activities, impactation of COVID-19 pandemic, and how they accumulated to imbibe the stress. Recently, there has been a growing boon interest in actually associating to leverage the ubiquitous sensing or sensor technologies for of course the main topic, i.e., mental health care applications and thus, giving them the allowance to start the continuous monetization of differential mental conditions such as depression, anxiety, stress, most important loneliness, working from home fluctuates the stress, sitting for a long duration of hours for scheduled meetings, online classes, increase in screen time, and more usage of smartphones and gadgets. Mental illness has no doubt a very serious inclination for not only the patients and their lifestyle but also for their families, friends, and society since it is difficult to cope with the basic implication of someone who is an association of close one having a mental illness [5, 20]. There have been an end number of implications of the new normal but since normal is changing and changing for the worse scenarios and managing the risk which is going to be the bitter truth of the near future. This isn't comfortable for most of us especially when we are trying to adjust and cope up with mental health. New normal for most of the public will generally mean what we as individuals need to go through this day or even this week. Feeling lonely or sad is also quite a common thing nowadays. Stay connected with one or the others to maintain a healthy foundation among the closed ones. Communication can help everyone to connect with family and friends and strengthen the foundation of love and affection. People can always call up individuals whom they haven't spoken to and surprise them. Discussion of happy events, common interests, exchange of cooking tips, share music, share old memories, recreate memories, and many more. Smartphones, laptops, and other technologically advanced devices have been considered to demonstrate that they have enough potential in providing the upcoming mental health interventions. Wearable technological devices such as smartphones, smartwatches, and fitness bands have a huge variety of embedded type sensors. They may include the communication devices such as WiFi and Bluetooth type and inertial sensors such as accelerometer and gyroscope. Physiological sensors such as heart rate and dermal activity type sensor and moreover the ambient sensor which metaphorically uses ambient pressure and thus stating the temperature are to basically name the few. Neuroscientists and clinicians around the world are using machine learning to implement the development for treating the plans and for patients to determine and identify some of the key markers for specular problems of mental health [6, 19]. One of the key features or we can say one of the benefits is that machine learning majorly helps to predict and analyze who may be at risk of one kind of

particular disorder. There is so much availability of data that we are now capable of compiling the data for the mental health professionals and analyzing the data so they may do their job better in a significant way. Making machine learning so helpful today is that in the past, the understanding of a diagnosis was casually based on the particular group averages may it be statistics overpopulations and what not. Machine learning gives the clinicians the opportunity for personalization, implementation, and exercisation on the group of data extracted. This paper or research work primarily hosted the surveys about the recent research works in the area of mental health which monitors the usage of the previously conducted surveys on the public, generally including the school-going children, college students and working professionals, using the data collected and machine learning. We focused on a prior basis for the research tasks about mental disorders/conditions which is inclusive of depression, anxiety, bipolar disorder, stress, etc. during and post lockdown [7, 18].

2 Background Study

Coronavirus disease or commonly known as COVID-19 is basically an infectious disease that is primarily caused by the recently discovered coronavirus.

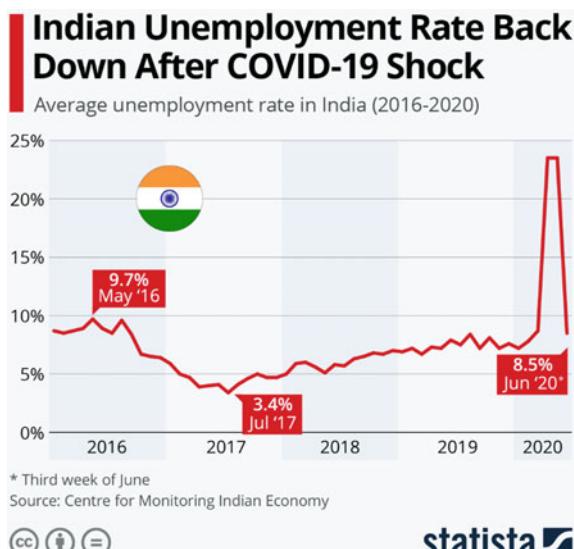
Most of the population in the world has been infected with the COVID-19 virus. People affected have experienced mild to moderate respiratory illness that may or may not recover without any kind of special treatment. People in the older age and the ones with any kind of medical problems that may include cardiovascular disease, diabetes, chronic respiratory disease, and cancer are more prone to get serious illness [8]. The best way to prevent or slow down the transmission is to stay well informed about this virus. We can be protected from this infection by washing our hands or using an alcohol-based rub frequently and not touching our face.

The COVID-19 virus spreads majorly through droplets of a person's saliva and discharge of the nose from an infected person when he/she coughs or sneezes. At this pandemic time, there are no such vaccines or any treatments for COVID-19. Although there are many ongoing clinical trials evaluating potential treatments globally. In India, the first lockdown came into being on 24 March 2020 which lasted for 21 days but sooner or later the public realized that this is a deadly virus and it is causing a pandemic worldwide, with the subsequent lockdowns that came into being, the general public was majorly affected as they happen to face the economical, social, health phobia, and majorly mental issues [9, 17] (Fig. 1).

In the above image, we see a tremendous rise in the unemployment rate. In India, people with different age groups faced different stumbling blocks to cope up with the increase in the number of COVID-19 cases, decline in employment, students that don't have an access to the Internet losing out on their studies, phobia to get infected with the deadly virus, mental torture, adaption to the new normal, and anxiety of being stuck at home (Fig. 2).

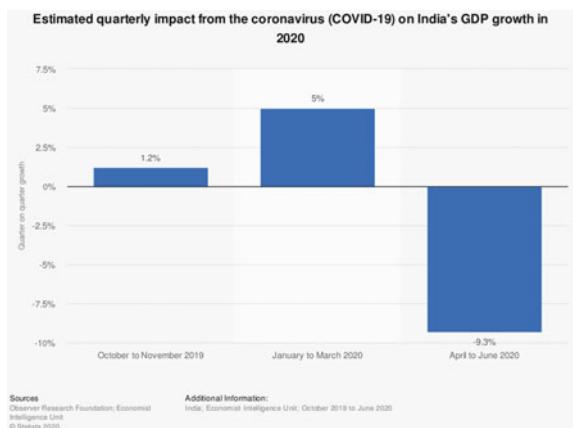
India's GDP decline is the lowest in the last six years which has affected the country at a huge level. The above graph clearly indicates the massive fall in India's

Fig. 1 Unemployment rate. This figure has been taken from a paper on “Indian Unemployment Rate Back Down After COVID-19 Shock”, by Katharina Buchholz, June 24, 2020 from Statista



statista

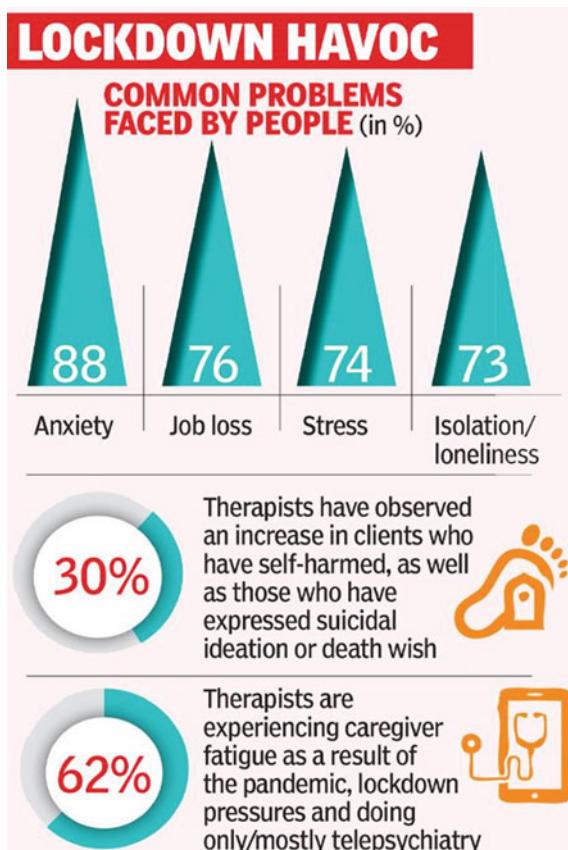
Fig. 2 India’s GDP. This figure has been taken from a paper on the Estimated quarterly impact of the coronavirus (COVID-19) on India’s GDP growth in 2020 on April 2020 from Statista



GDP from April to June 2020 [10, 16]. India’s upcoming population of young professionals may have difficulty securing jobs in the near future as the country’s economic growth touched a historic low in the April–June quarter 2020. Around 12.2 crore Indians who are working professionals have lost their jobs in April 2020 due to COVID-19. We can’t even imagine what mental breakdown these people have faced, with the unemployability on its peak, a series of anxiety, pressure, mental torture, and depression has led the working professionals affected with the new normal (Fig. 3).

In this analysis, we see the rise in mental health cases. Here in this paper, we have tended to impanel a systematic way to take a glance at the prior importance of mental health that should be given in our country. A systematic approach of machine

Fig. 3 Rise in mental health cases. This figure has been taken from a paper on “Spike in self-harm, suicide ideation amid COVID pandemic” by Priya Menon and Saranya Chakrapani, Jul 25, 2020 from TOI (Times of India)



learning has enabled you to implement and perform the analysis through supervised and unsupervised machine learning algorithms that are projected in Python. The role of Python in machine learning enables you to combine the apt and required power with a very clear and crisp type of syntax [11, 15]. It is also taken into consideration as an extended language for the applications that are scripted in other languages that may or may not need an easy-to-understand or any typical automation interface. It is the most preferred language for teaching curriculum and learning the basic concepts of MI (Machine Learning). Therefore, we projected our work on Python and generated the analysis and results. Machine learning is a major process of data analysis that helps to build an overview of the analytical model. It is referred to as a particular type in artificial intelligence which is particularly based on the system that has the ability to learn from the data to identify different types of patterns and make valuable implementations with the use of minimal human mediation.

```
In [4]: df = pd.read_csv('research_paper.csv')
df.head()
```

Out[4]:

	AGE GROUP	REGION	GENDER	Symptoms	DEPRESSION	STRESS
0	16-21	Delhi	F	2	Yes	Yes
1	22-28	Faridabad	M	0	No	No
2	29 - above	Noida	F	1	Yes	No
3	22-28	Bhopal	F	1	Yes	No
4	16-21	Delhi	M	1	No	Yes

Fig. 4 Dataset

3 Methodology

3.1 About the Dataset

Different people of different age groups and moreover different locations have gone through and post lockdown which occurred in India due to the spread of the disease COVID-19.

The dataset consisted of the following age groups 16–21 years, 22–28 years, and 29 years and above

The locations majorly covered were Faridabad, Delhi, Kolkata, Mumbai, Hyderabad, Ghaziabad, Chennai, Lucknow, Agartala, Patna Bhopal, Noida, and Gurugram. The dataset was divided in such a way that could clearly determine where and which age group were majorly affected. The algorithms were performed so as to determine the age value counts and mental health effects of the individuals regionwise. Two parameters of the dataset were created by a survey which was solely on the basis of different age groups as different mental states of mind in different age groups would rectify the scenario in a more broad way that could generate a clear picture of mental health that were considered in this paper are Depression and Stress [12] (Fig. 4).

3.2 Environmental Setup

Components of Python ML Ecosystem

In our research paper, we have used the Jupyter notebook which provides an interactive computational environment for the development of Data Science applications. They are probably known as IPython notebooks.

3.3 *Libraries Used*

Some of the libraries used

Numpy

Numpy is a very interactive and informative type of library that is considered to be very easy to use. It makes an attempt for solving the complex to complex mathematical implementations and scenarios with ease. The interface is primarily used and crafted for expressing the different images, different sound waves, and other types of binary raw streams taken as an array of the real numbers in N-dimensional.

SciPy

SciPy is considered as the machine learning library typically associated with the application developers and engineers. SciPy library consists of modules of optimization and linear algebraic equations, integration, and last but not least statistics. SciPy determines and provides an efficient numerical routine such as optimization, numerical integration, and many other types that use its specific submodules.

Pandas

It gives us the access to use the data structures of high-level in an organized way and a wide variety of tools for specific analysis. It has the power and capability to ease complex operations with data using one or maximum two commands. It has the inbuilt methodology for grouping, combining, and filtering, as well as theoretical time-series functionality. It also helps in operational functions such as Re-indexing, Iteration, Sorting, Aggregations, Concatenations, and Visualizations.

What Is Scikit-learn?

It is another type of Python library which is cumulatively associated with the other two types of libraries that are namely NumPy and SciPy. The cross-validation characteristic gives an ability to implement more than one metric. Cross-validation has numerous methodologies to check the apt accuracy of supervised models on a particular data which is not seen before in the scenario. Unsupervised learning algorithms consist of a huge number of algorithms in typical offerings—starting from clustering, factor analysis, principal component analysis to unsupervised neural networks. Feature extraction is basically used for the extraction of features from image concepts and textual information (e.g., Bag of words) [13].

4 Results and Discussion

4.1 Dataset Evaluation

For this survey, we have collected over 450 responses from different cities across India with a large impact and proportion coming from educated youth, i.e., the age group 22–28. According to the survey's reports, individuals were recorded more than one cause for a shooting rise in mental health problems (Figs. 5 and 6).

According to the survey's report, several responses were recorded across 13 cities of which most cases were recorded in Delhi (Fig. 7).

According to the survey's report, the majority of the population responded to suffering from depression. Individuals were recorded more than one cause for a shoot in the mental health problems (Fig. 8).

The pandemic caused a spike in stress levels among different age groups due to various reasons. For the younger generations, a slight shift in terms of education, and for particularly those in their 20 and 30 s, job crunches and losses primarily added to problems of worry and tension (Fig. 9).

The report also suggested an apparent skewed difference between men and women. It was taken into the observation that mental well-being was low scored in males (Fig. 10).

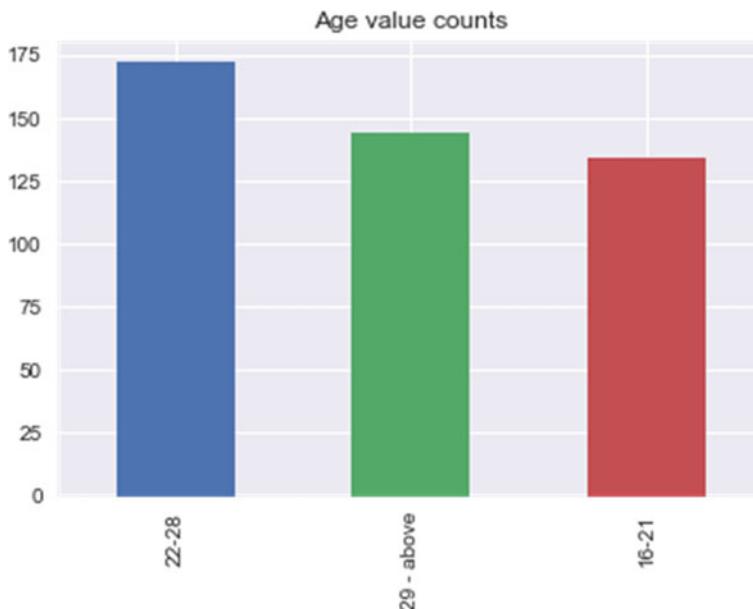


Fig. 5 Data collected from different age groups through study from survey

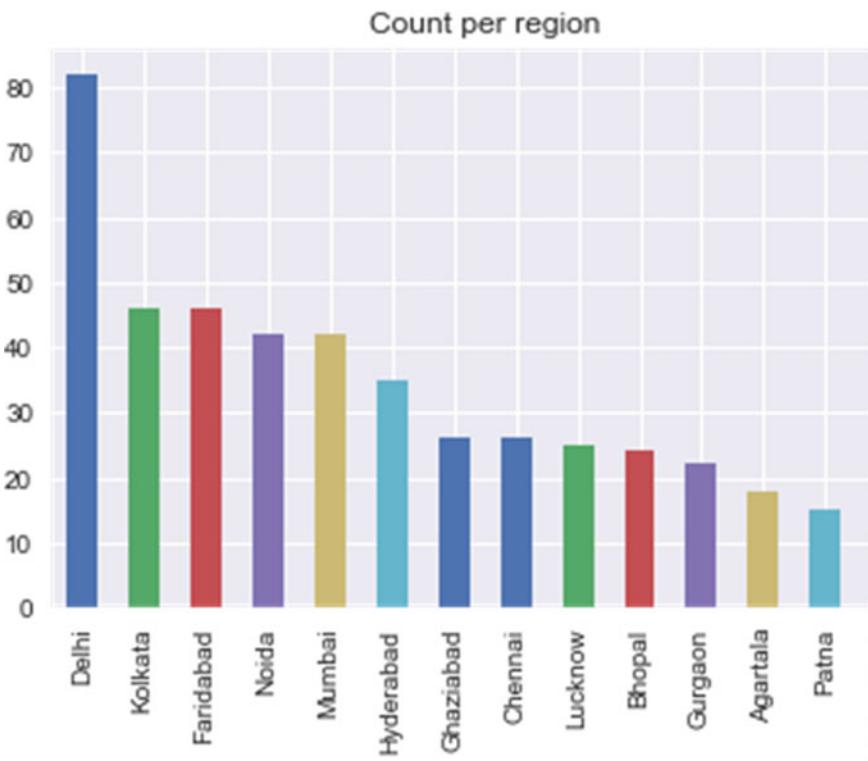


Fig. 6 Mental health data count regionwise

The majority of the Indians are experiencing either of the two symptoms (stress/depression) because of the uncertainty and looming financial crisis during the lockdown (Fig. 11),

22–28 age group showing the highest count followed by 29 and above age group and then 16–21 age group, and the males of every age group showing the highest stress count (Fig. 12).

There are more number of people suffering from depression, the majority being the male population, as per the data collected from the survey analysis (Fig. 13).

There are more number of people suffering from stress, the majority being the male population, as per the data collected from the survey analysis.

4.2 Heat Map Comparison

The heat map is a one on one comparison between each parameter, it is a data visualization technique that uses color as a tool for study. The above graph contains a range of colors from dark green to light yellow, the former represents the standard

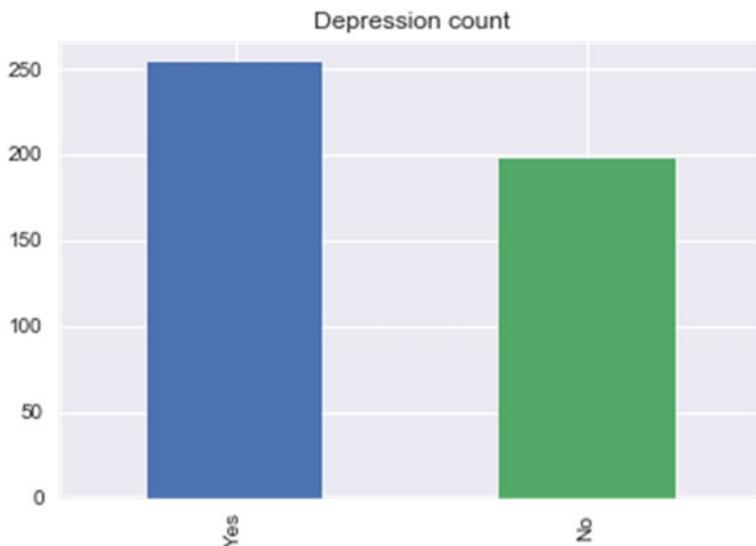


Fig. 7 The analysis of the survey shows a major depressive episode among the different age groups

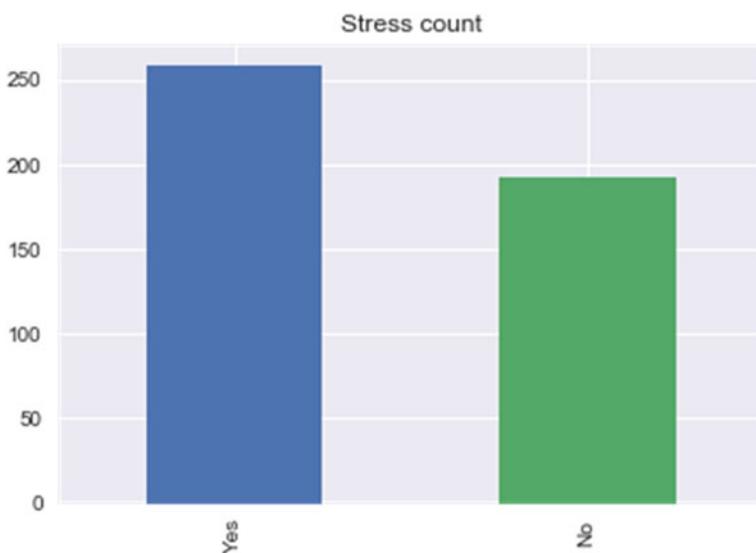


Fig. 8 The analysis of the survey shows a major stress count among the different age groups

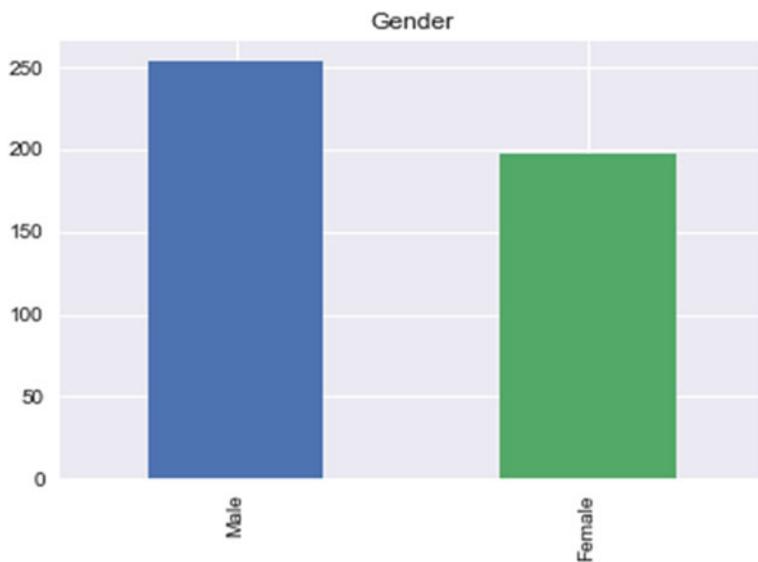


Fig. 9 Depression and stress count among the males and females of different age groups

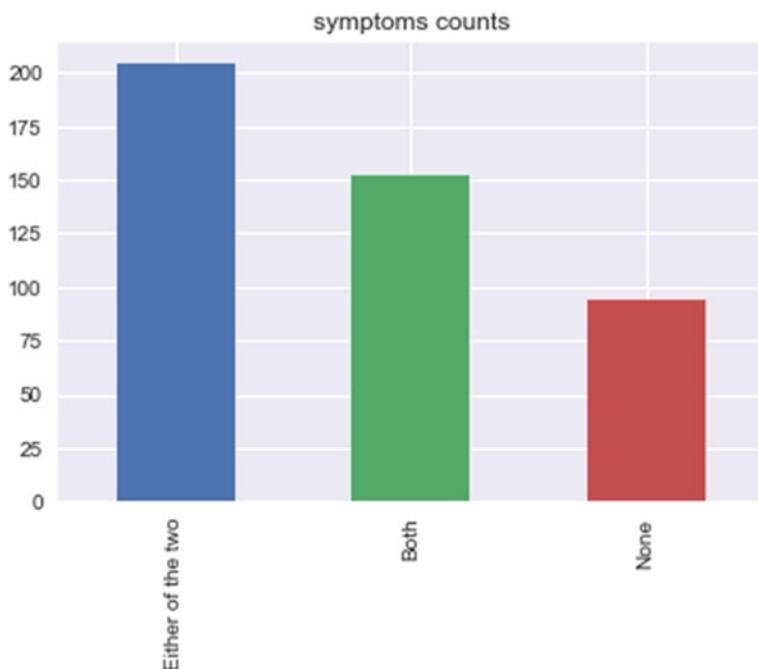


Fig. 10 Symptoms count among the different age groups

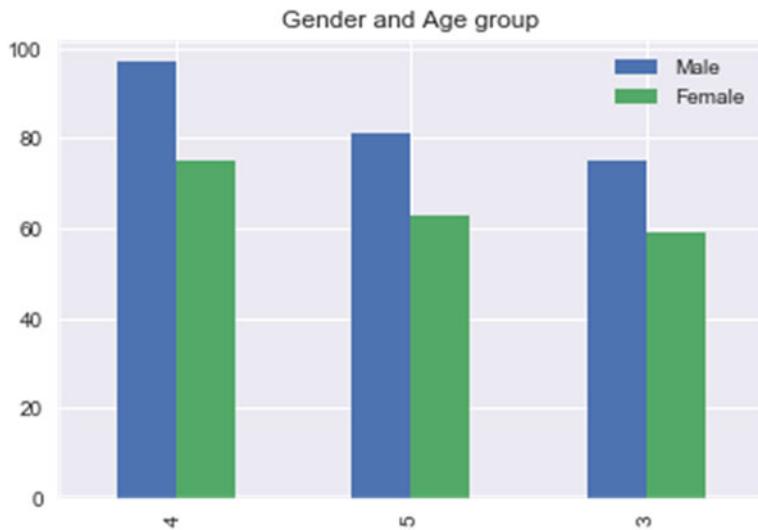


Fig. 11 Stress count of the different age groups and gender

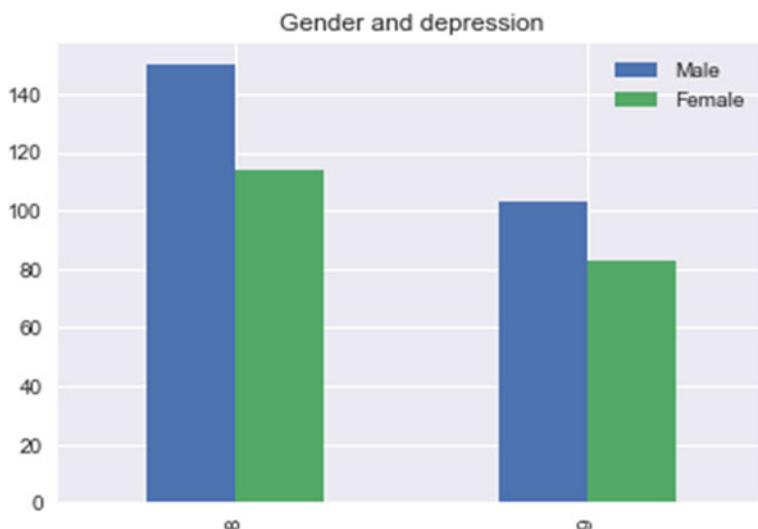


Fig. 12 Graph representing Depression count versus Gender

parameters and the latter represents non-standard parameters. Standard parameters are used as it is but the non-standard parameters were processed later for analysis (Fig. 14).

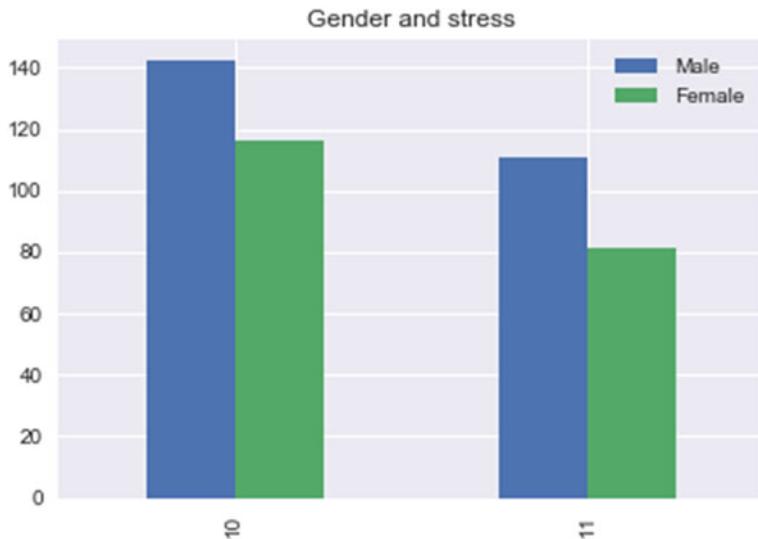


Fig. 13 Graph representing stress count versus gender

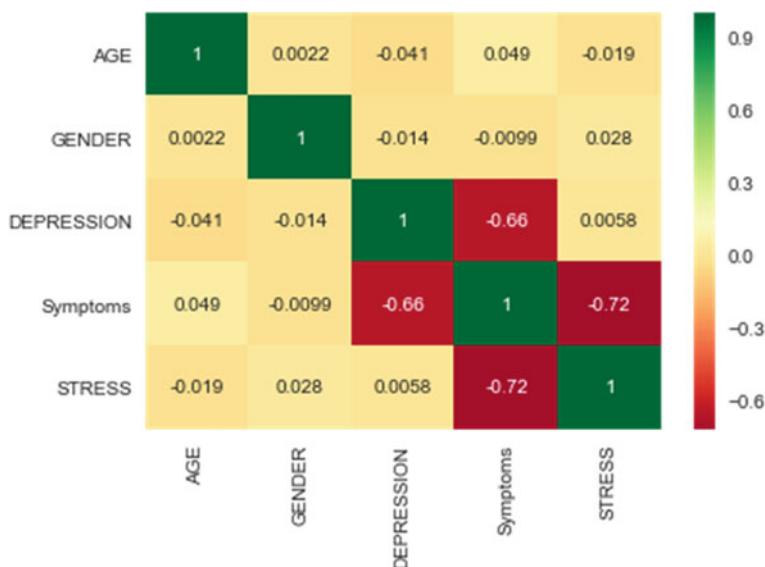


Fig. 14 One on one comparison between each of the parameters

```

plt.scatter(Y.AGE, Y.Symptoms, color = "purple", label = "Female", alpha = 1.0)
plt.scatter(N.AGE, N.Symptoms, color = "orange", label = "Male", alpha = 1.0)
plt.xlabel("AGE")
plt.ylabel("Symptoms")
plt.legend()
plt.show()

```

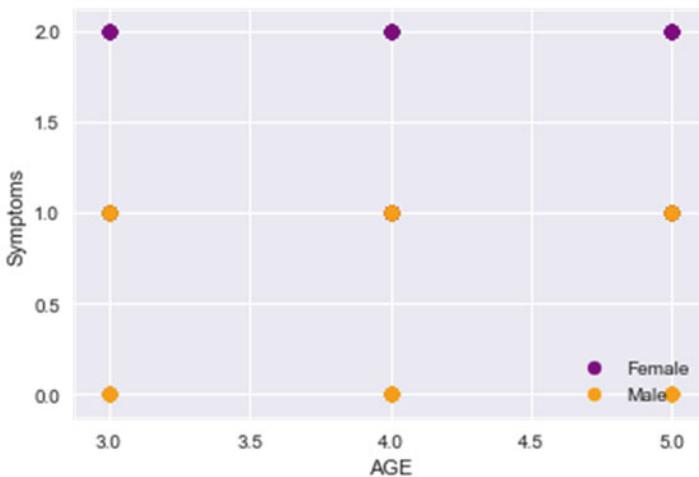


Fig. 15 The graph shows a dot plot of the number of males and females with respect to symptom and age

4.3 KNN Result

In this research paper, the supervised learning algorithm used is K-nearest neighbor. KNN is a very simple and basic algorithm that runs on the principle of closely related parameters. What we mean by that is the parameters which are closely related to each other are found in close proximity (Fig. 15).

In our research paper, we have used 3 age groups that are 16–21, 22–28, and 29 and above represented as 3, 4, and 5, respectively. The symptoms are also represented as 0, 1, and 2 for none, single, and both the symptoms. Symptoms here are depression and stress. So, we can clearly see that for 0 or 1 symptom in all three age groups, we see the majority as male and for both symptoms in all three age groups, we see female as the majority [14] (Fig. 16).

5 Future Scope

The pandemic has created a hustle-bustle all over the world which has taken almost a year to settle down. People are trying to cope up with their mental illness especially

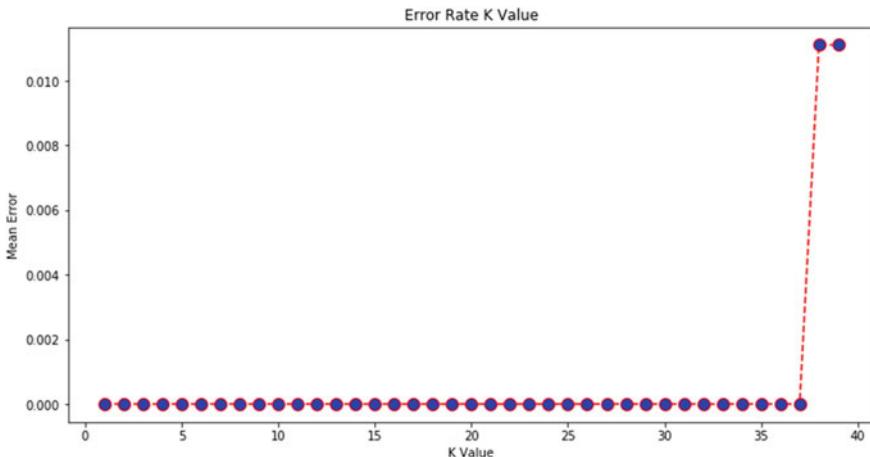


Fig. 16 Here we have plotted the above graph to analyze further by the measures of error

the age groups mentioned above. Stating the future of this research paper is significant to determine if any such situation may prevail again then proper preventive measures and solutions to deal with these types of mental illness will be considered with utmost importance. Analyzing the results so obtained in this research paper, we can clearly figure out the appropriate measures that can be taken into consideration for any such dilemma in the near future.

Artificial Intelligence and machine learning are two influential communities that are investing in mental health care, research, and studies. The future of machine learning in evaluating mental health seems very promising and is proactively looking for the right direction.

6 Conclusion

As data scientists, practitioners, and researchers are actively investing time, money, and energy for mental health scenarios with the right approach of directions, analysis and such studies of data with the use of machine learning mark the baseline for the future aspects and any kind of future developments, studying the perspectives of the general mental health of people, determining the apt ratios, and quantitative research in any field. According to our survey, we evaluated the stress and depression count among a certain age group, which was found to be the highest in the age group (22–28). Therefore, we conclude that we should take the right approach with utmost spirit in forming the introduction of AI into the mental cycle by taking help from our technological analysis and studies with the expertise of data analysis and similarly collaborating with computational scientists, as well as other types of experts, to help

and transform the mental health practice and improve the care for patients not only physically but mentally also.

References

1. Darien Miranda, Marco Calderon, Anxiety detection using wearable Monitoring, 1 November 2014 https://www.researchgate.net/publication/288492542_Anxiety_detection_using_wearable_
2. Outlook—The new Scroll article | 31 March 2020 <https://www.outlookindia.com/newsscroll/dont-drink-alcohol-to-cope-with-lockdown-ministry/1786251>
3. I. Sharma, A. Agarwal, A. Saxena, S. Chandra, Development of a better study resource for genetic disorders through online platform. *Int. J. Inf. Syst. Manag. Sci.* **1**(2), 252–258 (2018)
4. S. Mohagaonkara, A. Rawlani, P. Srivastavac, A. Saxena, HerbNet: Intelligent knowledge discovery in MySQL database for acute ailments, in *Proceedings of the 4th International Conference on Computers and Management (ICCM) 2018ELSEVIER-SSRN* (ISSN: 15565068), pp. 161–165
5. S. Shuklaa, A. Saxena, Python based drug designing for Alzheimer's disease, in *Proceedings of the 4th International Conference on Computers and Management (ICCM) 2018ELSEVIER-SSRN* (ISSN: 15565068), pp. 2024
6. A. Agarwal, A. Saxena, Comparing machine learning algorithms to predict diabetes in women and visualize factors affecting it the most—a step toward better healthcare for women, in *Proceedings of the International Conference on Innovative Computing and Communications*, https://doi.org/10.1007/978-981-15-1286-5_29, 2019
7. A. Saxena, S. Chandra, A. Grover, L. Anand, S. Jauhari, Genetic variance 13 study in human on the basis of skin/eye/hair pigmentation using apache spark, in *Proceedings of the International Conference on Innovative Computing and Communications*, https://doi.org/10.1007/978-981-15-1286-5_31, 2019
8. L. Miner et al., *Practical Predictive Analytics and Decisioning Systems for Medicine: Informatics Accuracy and Cost-Effectiveness for Healthcare Administration and Delivery Including Medical Research* (Academic Press, Cambridge, 2014)
9. D.D. Luxton (ed.), *Artificial Intelligence in Behavioral and Mental Health Care* (Elsevier Inc., Amsterdam, 2015)
10. T. Hahn, A.A. Nierenberg, S. Whitfield-Gabrieli, Predictive analytics in mental health: applications, guidelines, challenges and perspectives. *Mol. Psychiatry* **22**(1), 37–43 (2017)
11. R.V. Bijl, A. Ravelli, G. Van Zessen, Prevalence of psychiatric disorder in the general population: results of The Netherlands Mental Health Survey and Incidence Study (NEMESIS). *Soc. Psychiatry Psychiatr. Epidemiol.* **33**(12), 587–595 (1998)
12. World Health Organization, Mental health: a call for action by world health ministers (Geneva, World Health Organization, Department of Mental Health and Substance Dependence, 2001)
13. M. Funk, Global burden of mental disorders and the need for a comprehensive, coordinated response from health and social sectors at the country level. http://apps.who.int/gb/ebwha/pdf_files/EB130/B130_9-en.pdf. Accessed 20 Feb 2016, 2016
14. A. Drapeau, A. Marchand, D. Beaulieu-Prévost, Mental illnesses- understanding, prediction and control. *Epidemiol. Psychol. Distress* (2012). <https://doi.org/10.5772/1235>
15. A. Agarwal, A. Saxena, Malignant tumor detection using machine learning through Scikit-learn. *Int. J. Pure Appl. Mathem.* **119**(15), 2863–2874, ISSN: 1314–3395 (2018)
16. A. Agarwal, A. Saxena, Comparing machine learning algorithms to predict diabetes in women and visualize factors affecting it the most—a step toward better health care for women, in *Proceedings of the International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing*, vol. 1087 (Springer, Singapore, 2020), pp. 339350

17. A. Saxena, N. Kushik, A. Chaurasia, N. Kaushik, Predicting the Outcome of an election results using sentiment analysis of machine learning, in *Proceedings of the International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing*, vol. 1087 (Springer, Singapore, 2020), pp. 503–516
18. A. Agarwal, A. Saxena, Analysis of machine learning algorithms and obtaining highest accuracy for prediction of diabetes in women, in *Proceedings of the 6th International Conference on Computing for Sustainable Global Development (INDIACom)*, (New Delhi, India, 2019), pp. 686–690
19. S. Mohagaonkar, A. Rawlani, A. Saxena, Efficient decision tree using machine learning tools for acute ailments, in *Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)* (New Delhi, India, 2019), pp. 691–697
20. Dubey, Aman Kumar, R. Krishna, S. Aravind, Mohagaonkar, Sanika, Saxena, Ankur, Prediction of coronavirus outbreak based on cuisines and temperature using machine learning algorithms (May 23, 2020). Available at SSRN: <https://ssrn.com/abstract=3608767> or <http://dx.doi.org/> <https://doi.org/10.2139/ssrn.3608767>
21. S. Mohanty, R. Sharma, M. Saxena, A. Saxena, Heuristic approach towards COVID-19: big data analytics and classification with natural language processing, in *Data Analytics and Management. Lecture Notes on Data Engineering and Communications Technologies*, ed. by A. Khanna, D. Gupta, Z. Pólkowski, S. Bhattacharyya, O. Castillo, vol. 54 (Springer, Singapore, 2021). http://doi.org/443.webvpn.fjmu.edu.cn/https://doi.org/10.1007/978-981-15-8335-3_59
22. S. Mohanty, A. Mishra, A. Saxena, Medical data analysis using machine learning with KNN, in *Proceedings of the International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing*, ed. by D. Gupta, A. Khanna, S. Bhattacharyya, A. Hassanien, S. Anand, A. Jaiswal, vol. 1166. (Springer, Singapore, 2020). http://doi.org/443.webvpn.fjmu.edu.cn/https://doi.org/10.1007/978-981-15-5148-2_42
23. M. Saxena, A. Deo, A. Saxena, mHealth for mental health, in *Proceedings of the International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing*, ed. by D. Gupta, A. Khanna, S. Bhattacharyya S, A.E. Hassanien, S. Anand, A. Jaiswal, vol. 1165 (Springer, Singapore, 2020). http://doi.org/443.webvpn.fjmu.edu.cn/https://doi.org/10.1007/978-981-15-5113-0_84
24. M. Saxena, A. Saxena, Evolution of mHealth Eco-System: a step towards personalized medicine, in *Proceedings of the International Conference on Innovative Computing and Communications. Advances in Intelligent Systems and Computing*, vol. 1087 (Springer, Singapore, 2020), pp. 351370

An Efficient Approach to Predict Fear of Human's Mind During COVID-19 Outbreaks Utilizing Data Mining Technique



Diti Roy, Tamal Joyti Roy, Ikbal Mahmud, and Nasif Alvi

Abstract COVID-19 has severely affected the world health and economic sector. It has also affected the psychological behavior of people of every ages. That's why this study has been conducted, "Detecting Fear of COVID-19". A wide variety of data from different ages including student, jobholder, doctor, businessmen, unemployed person, and others is collected for conducting this study. We have collected 553 instances to complete this analysis. By using this data, we have constructed a detection system which help us to detect the fear of COVID-19. We have constructed a machine learning classifier by using ten machine learning algorithms and their features technique. Finally, two machine learning algorithms have been used to identify the fear of human's mind during Covid-19 outbreaks. One is LogitBoost and the other one is Random Forest algorithm. With the assistance of tenfold-cross validation, we have measured the validity of data which is collected by us, whereas performance matrix has helped us to report the evaluation of data. This evaluation report has shown us the accuracy and effectiveness of constructing a model to detect the fear of COVID-19. We have gotten the final result, that is, 70.34% by using LogitBoost algorithm. Our main goal is to identify the fear of COVID-19, as many people were afraid of this virus.

Keywords COVID-19 · Corona virus · Fear · Classifications · Prediction · LogitBoost · Random Forest

1 Introduction

To deal with the upcoming threat of coronavirus, fear is an important emotion which distributes to mobilize energy. However, to avoid maladaptive situation, fear should be calculated to the actual threat [1]. The coronavirus disease, 2019 (COVID-19) outbreak, is caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), which was originated in Wuhan, China, in December 2019 [2]. Though

D. Roy (✉) · T. J. Roy · I. Mahmud · N. Alvi
North Western University, Khulna, Bangladesh

human coronaviruses have been appeared for many years, but COVID-19 is more lethal than others because of its wide global sparked panic among people. That's why since early 2020, COVID-19 is considered as pandemic and every government started to follow the rules and regulations to manage this crisis. In order to control this virus, the government had introduced the term **Lockdown**, which was unimaginable a few days ago. According to Dawson and Moghaddam [12], it has ultimately reduced working hours of labor, school time for students, entertaining time for wanderlust and spectator, shopping time for buyer, get-together with family members and friend, etc. Which is ultimately symbolized as anxiety, depression, and loneliness. As it is contagious and people don't have natural immunity to this new strain of coronavirus, a large number of people all over the world are already affected by it [3]. Though a lot of steps have been taken by the government of different countries to control the virus, but still the situation is being deteriorated day by day with huge amount of economic loss such as unemployment, inflation, mismanagement of demand and supply sector, as well as a huge number of death. As people are not used to this type of situation they are facing different type of problems such as unemployment problems, lower status and lack of money. Along with these problem, limiting social contracts increases depression and loneliness in people. According to Groarke et al. [13], the impact of Covid-19 is severe with respect to public mental health. At least 25% people are suffering from loneliness in the form of depression, disturbance in sleep, lack of attention in study and lack of power in controlling emotion, etc. As we know that data mining is a widespread tool to predict something by using data, this study has been conducted with a view to predict the fear of Covid-19 by using data mining approach.

The rest of this research study is organized as follows. In Sects. 2 and 3, the existing works and working procedure have been explained with a through analysis of the algorithms, respectively. In Sect. 4, the outcome of this analysis has been clarified with the impulsion to justify the novelty of this exploration work. Finally, this research paper is terminated with Sect. 5.

2 Related Works

To fulfill the unknown information in regard to corona virus to researchers, doctors and policy makers, many researchers are conducting study on corona virus using different techniques including data mining. In this part, we have discussed various studies associated with COVID-19. To detect COVID-19, RT-PCR method is popular all over the world but to find this result it takes long time. To overcome this problem, CT images can be used to detect COVID-19. Therefore, Ouyang et al. [5] focused on the overview of dual sampling attention network for the diagnosis of COVID-19 from Community Acquired Pneumonia (CAP) in chest CT images. They had evaluated the method upon largest multi-center CT data for COVID-19 which was collected from 8 hospitals. To conduct this study, 2186 CT scans for 1588 patients were collected for a fivefold cross validation. Again for evaluating the performance and generalization

ability, independent testing of 2796 CT scans from 2057 patients were collected by them. Firstly, they suggested a model with 3D convolution network to identify infectious region in lungs, which would help to make decision of diagnosis. They had also proposed a dual sampling attention network to classify the COVID-19 and CAP infections. In their experiment, they had found that COVID-19 cases were more severe than CPA cases. They had used VB-Net toolkit in Training Validation stage to show the distribution of ratios between infectious region and lungs. Finally, they had found that their algorithm identified the COVID-19 images with 0.944 the area under receiver operating characteristic curve (AVC) value, 87.5% accuracy, 86.9% sensitivity, 90.1% specificity, and 82% F1-score. On the basis of learning theory, Duffey et al. [4] had tried to predict a recovery rate of infections in terms of infectious disease such as COVID-19. Researchers had collected data from different parts of the world to construct a well-defined prediction on retrieval trend and time needed for achieving lowest rate of infection which would be used to determine the time when the pandemic would be under control. It also analyzed how different types of countermeasures such as social distancing, using mask, sanitization or human adaptability by learning from experience will help to reduce infection and death rate. When they observed the infection rate of different countries by using IHME projection model they had found the learning curve theory for data fitness. This analysis would help countries to contrast different types of countermeasure to manage this pandemic. After analyzing, they had finally recommended to limit personal distance to overcome the problems of getting infected. Heart disease is one of the lethal disease of the current era. Though medical authority is rich in information, but still it lacks knowledge. Not only adequate treatment is essential, but also treatment is needed on time. That's why Raju et al. [6] conducted this study to find the efficacious treatment for heart disease by using data mining technique. To gain a perfect accuracy, authors used different algorithms, which were used to data mining classification such as Neural network, Naive Bayes, Genetic algorithm, Decision Tree, and Support Vector Machine (SVM). They finally found that Support Vector Machine technique produced the best result with the accuracy of 99.3%. Sonu et al. [7] conducted this study to predict Parkinson's disease of a person using voice record with assistance of data mining technique. To complete this study, they had used different techniques such as KNN, Logistic Regression, Decision Tree, LDA, NB, and SVM. The voice of patients was recorded and converted into voice attributes like jitter and shimmer with the help of PRAAT scripts. Among these different techniques, Decision Tree was better as it provided implicit feature selection. After evaluation, they found that Decision Tree helped to identify Parkinson's disease with the accuracy of 100% without any feature, on the contrary, when they used feature, they got an accuracy of 88 to 94% only. To counter every disease, relevant immune pathogenesis is most important. That's why Tay et al. [8] introduced different techniques for SARS-CoV-2 infection and immune pathogenesis for COVID-19. In this study, Researcher had given an overview on interaction of SARS-CoV-2, as well as how dysfunctional immune system could help disease progression, which would ultimately help to produce a vaccine against lethal COVID-19. They found that, incubation period for COVID-19 is 4–5 days and 97.5% people showed symptoms within 11.5 days. They

also found that people who infected by COVID-19 showed the symptoms of fever, dry cough, difficulty in breathing, muscle pain, dizziness etc. They showed that sex hormone may influence fatality rate of COVID-19 as data indicated that the death rate of male was 2.8% which was higher than the death rate of female, which was 1.7%. From the analysis, they had suggested that, controlling the dysfunctional immune system and different types of ongoing therapies such as corticosteroids, antagonist tocilizumab would be helpful in future. The most important factor they have found is that T-cell is most important for eliminating virus and developing a vaccine.

3 Methodology

Data Mining is the PC helped cycle of extricating information from huge measure of data. In different words, information mining infers its name as Data + Mining, a similar manner by which mining is done in the ground to locate a significant metal. Information mining is done to discover important data in the dataset. We have gathered numerous answers from practically 553 individuals.

1. Data Collection
2. Data Preprocessing
3. Data Handling
4. Classifier Selection
5. Tools and Techniques

For completing this analysis, we have imported 553 collected data and added 25 features. Then, we have preprocessed the data with several techniques and applied two machine learning algorithms: Logit Boost and Random Forest Algorithm for this study. Finally, we have visualized the data and compared the algorithm where we get the result.

3.1 *Data Collection*

For analysis, 553 instances of data were collected that was filled by different profession of people like doctors, jobholders, students, businessmen, and others (Table 1).

3.2 *Data Preprocessing*

Every information can have some unimportant and nonessential data which may affect the analysis. To overcome this problem, data cleaning is done. Managing missing information, loud information, and so forth is included in this part. Overlook

Table 1 Features (Questionnaires) list

Features	Subcategory	Distribution (%)
Age	Minimum:20	
	Highest:75	
Occupation	Student	80.47
	Jobholder	11.39
	Businessman	2.53
	Doctor	1.80
	Others	3.81
Following the home quarantine procedure	Yes	88.97
	No	11.03
Enjoying home quarantine	Yes	47.01
	No	52.99
Complete lockdown is the only solution to overcome the COVID-19 problem	Yes	55.88
	No	44.12
All the affected people are maintaining quarantine properly	Yes	14.83
	No	85.17
Following all the news of COVID-19	Yes	85.36
	No	14.64
People in social media exaggerate the news of COVID-19	Yes	68.80
	No	31.20
Anyone can get well soon from COVID-19 by adequate medication	Agree	62.57
	Disagree	37.43
Drinking alcohol & smoking can help COVID-19 patients	Yes	13.56
	No	86.44
Heat can diminish this virus	Yes	46.83
	No	53.17
Buy fewer goods during the COVID-19 pandemic	Yes	71.06
	No	28.94
More than 75% of patients in our country getting treatment at home	Yes	72.51
	No	27.49
Wearing mask when you go outside	Yes	93.13
	No	6.87
Using mask is rumor while outing outside	Yes	33.63
	No	66.37
If government has eased lockdown completely you behave in the same way before lockdown	Yes	45.20
	No	54.80

(continued)

Table 1 (continued)

Features	Subcategory	Distribution (%)
Lack of adequate testing, the affected people number seems to be lower than the actual number	Yes	85.35
	No	14.65
Maintaining a healthy relationship with your family during the lockdown	Yes	90.96
	No	9.04
The COVID-19 pandemic will stay for a year or two	Yes	79.75
	No	20.25
If the vaccine will not available, move freely in the future	Yes	27.30
	No	72.70
Easing lockdown can increase the spread of COVID -19 in the future	Yes	80.10
	No	19.90
Viruses can't do much harm after developing antibodies in human	Yes	77.75
	No	22.25
Have any fear of COVID-19 (class)	Yes	65.82
	No	34.18

the tuples, this methodology is perfect just when we have a large amount of data and different qualities are not available within tuples. To fill the absence quality, there are different approaches available which will help to finish this work. In this study, we have intended to manage the missing qualities physically, through property mean or most likely worth.

3.3 Data Handling

For Data Handling, we have used the following approaches:

Binning Method To manage noisy data, Binning method is an important tool in Data Mining strategy. In this method, firstly the data are sorted into different sections and then this organized data are distributed into a number of buckets. Each sorted data is dealt with independently. With the assistance of its mean or limit esteems, one can supplant all information in a portion to finish this assignment.

Relapse To make the information smoother, relapse function can be used. This function can be used with either direct (having one autonomous variable) purpose or various (having different free factors) purpose.

Grouping This technique accumulated comparative information in a group. The anomalies might be undetected or it will fall outside the bunches.

Table 2 Parameters

Name	Type	Value
Base estimator	Object	None
N estimators	Int	50
Wright trim quantile	Float	50
Max response	Float	4
Learning rate	Float	1
Bootstrap	Bool	False
Random state	Int	None

3.4 Classifier Selection

For classification, we have used two algorithms. One is Logit Boost and the other one is Random Forest Algorithm. The parameters of the algorithms were shown in Table 2.

LogitBoost Logit Boost is known as a boosting classification algorithm. Logit Boost (LB) and AdaBoost(AB) algorithm are almost same as they help to calculate additive logistic regression. The discrepancy between them is exponential loss is minimized by AB algorithm, whereas the logistic loss is minimized by LB algorithm [9]. A Logit Boost classifier is a meta-estimator that fits an additive model minimizing a logistic loss function [10].

Random Forest To construct a large number of individual decision trees, random forest works as an ensemble. A class prediction is spit out by every individual tree lying in the random forest, which is used in the prediction of this model [11]. The use of Random Forest is easy. We have used this algorithm to identify best results. The necessities for random forest are:—Some actual signals in our features are needed, so that models built using those features perform better than random guessing. —The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

Tools and Techniques Different types of tools and techniques have been used in this study including Classification, Clustering, Regression, Association rules, Outer detection, Sequential Patterns, and prediction. Due to lack of advance technical knowledge, sometimes it is difficult for us to operate some analytics software.

4 Outcomes

For class **Do you have any fear of COVID-19**, we perform Logit Boost and Random Forest and get the accuracy of the class and confusion matrix.

Precision is the ratio between the true positive and all positive. For our problem statement that is the measure of people that we correctly identify having fear out of all the samples we collected.

$$\text{Precision} = \frac{\text{True Positive (TRPV)}}{\text{True Positive (TRPV)} + \text{False Positive (FLPV)}}$$

The recall is the measure of our model correctly identifying true positives. Such as all the people we have found who has fear, recall expressed how many we correctly select as having fear. Mathematically

$$\text{Recall} = \frac{\text{True Positive (TRPV)}}{\text{True Positive (TRPV)} + \text{False Negative (FLNV)}}$$

So we have found our accuracy in following ways

$$\text{Accuracy} = \frac{\text{TRPV} + \text{TRNV}}{\text{Positive} + \text{Negative}}$$

To complete this analysis, we have used the class **Do you have any fear of COVID-19**. In this class, Logit Boost and Random Forest algorithm are used. Accuracy, FP Rate, TP rate Precision, Recall, F-Measure, MCC, Roc Area, and PRC Area are found by using both the algorithms, which is shown in Table 3. Around 74.9% percent of TP Rate, 71.3% of FP Rate, 66.2% of Precision, 74.9% of Recall, 67.7% of F-Measure, 7.2% of MCC, 58.6% of Roc Area, and 68.8% of PRC Area was found by using Logit Boost algorithm. Around 69.6% percent of TP Rate, 45.8% of FP Rate, 67.5% of Precision, 68.6% of Recall, 66.8% of F-Measure, 28.6% of MCC, 68.0% of Roc Area, and 69.1% of PRC Area was found by using Logit Boost algorithm. Finally, 70.34% of Accuracy was predicted by using Logit Boost algorithm and 69.620% of Accuracy was identified by using Random Forest algorithm, shown in Figs. 2, 3 and 4. The best result is identified by using Logit Boost algorithm.

Table 3 Outcome of the classifiers

Evaluation matrix	Name of the algorithms	
	Logit boost	Random forest
Correctly classified instances	414	385
Incorrectly classified instances	139	168
Accuracy	70.34%	69.620%
TP Rate (Weighted Avg)	74.9%	69.6%
FP Rate (Weighted Avg)	71.3%	45.8%
Precision (Weighted Avg)	66.2%	67.5%
Recall (Weighted Avg)	74.9%	68.6%
F-Measure (Weighted Avg)	67.7%	66.8%
Matthews correlation coefficient(MCC) (Weighted Avg)	7.2%	28.6%
Roc area (Weighted Avg)	58.6%	68.0%
PRC area (Weighted Avg)	68.8%	69.1%

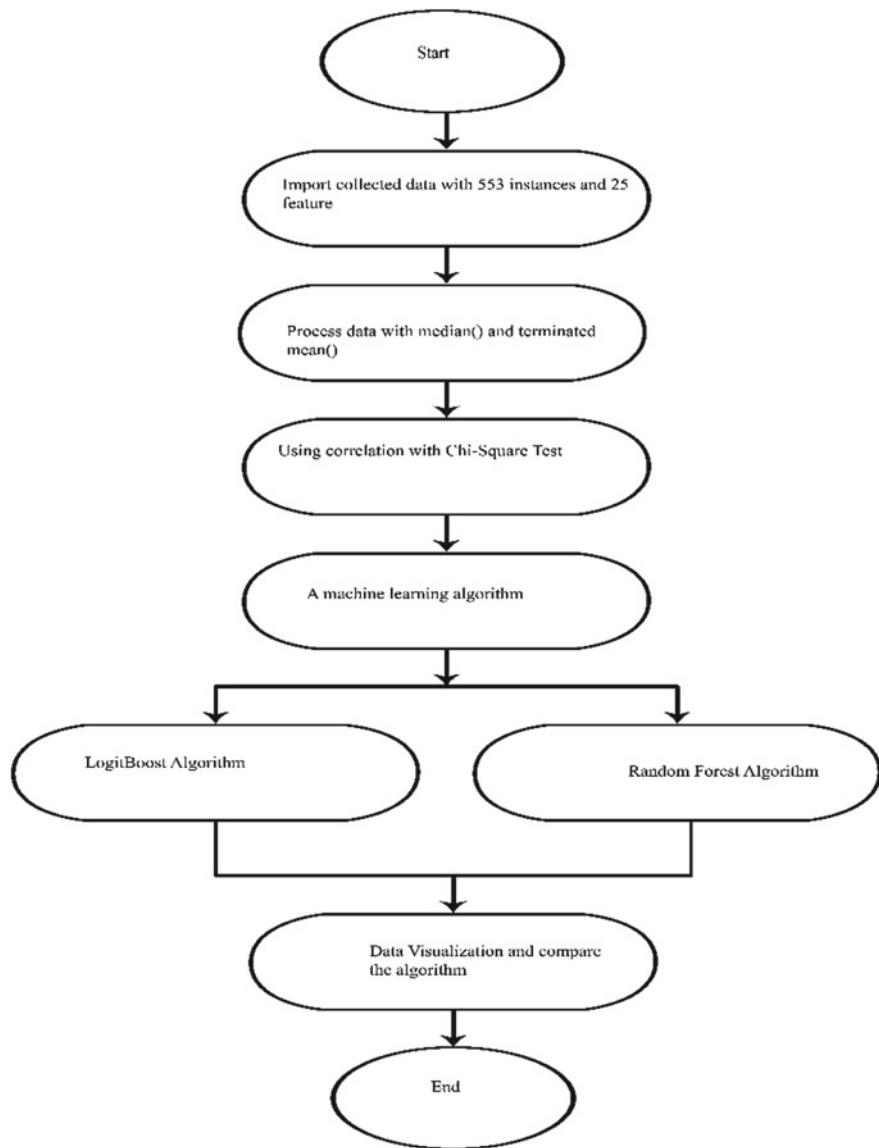


Fig. 1 Overall working procedure

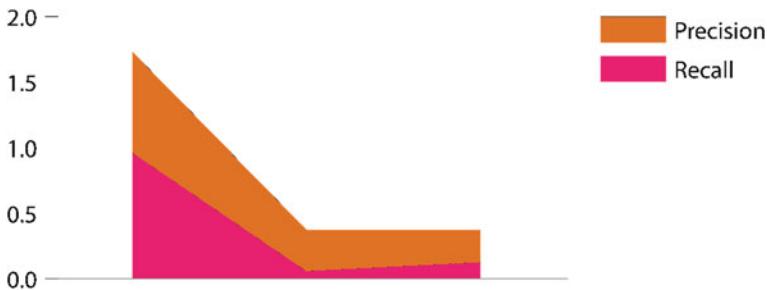


Fig. 2 Graphical representations of accuracy for class 1



Fig. 3 Graphical representation of PR curve using logit boost algorithm

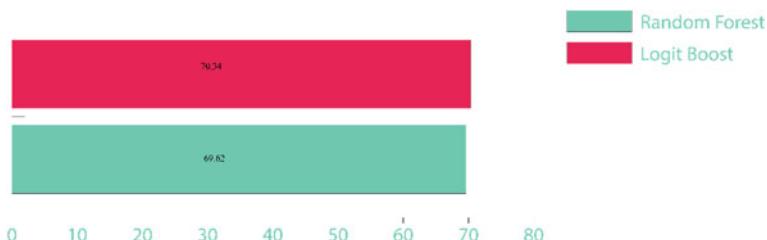


Fig. 4 Graphical representation of PR curve using random forest algorithm

5 Conclusion

Nowadays, many people are directly or indirectly affected by COVID-19. They have been suffering from different problems including health, economic crisis due to Corona virus. In this study, we have proposed a model which is able to predict fear of human's mind of all ages during COVID-19 outbreaks. Firstly, 553 instances are collected from different ages of people. The class "Do you have any fear of COVID-19" has been used for this study. We have used ten machine learning algorithm and their feature extraction for classification. For testing this collected data, tenfold cross-validation has been used. To achieve best result, different types of machine

learning algorithms have been used. Finally, we are able to show the best result by using LogitBoost and Random Forest algorithm. LogitBoost algorithm gives 70.34% accuracy for this class and Random Forest gives 69.62% accuracy for this class. Within two algorithms, we have gotten the best result by using LogitBoost algorithm to identify the fear of COVID-19. To know more about the trend of COVID-19 and how it influences the fear of people, more study needs to be added.

References

1. G. Mertens, L. Gerritsen, S. Duijndam, E. Saleminck, I. Engelhard, Fear of the coronavirus (COVID-19): Predictors in an online study conducted in March 2020. *J. Anxiety Disord.* **74**, 102258 (2020)
2. S. Ren, R. Gao, Y. Chen, Fear can be more harmful than the severe acute respiratory syndrome coronavirus 2 in controlling the corona virus disease 2019 epidemic. *World J. Clin. Cases* **8**(4), 652–657 (2020). Available at: <https://doi.org/10.12998/wjcc.v8.i4.652>
3. K. Goyal, P. Chauhan, K. Chhikara, P. Gupta, M. Singh, Fear of COVID 2019: first suicidal case in India ! *Asian J. Psych.* **49**, 101989 (2020)
4. R.B. Duffey, E. Zio, Analysing recovery from pandemics by learning theory: the case of CoVid-19. *medRxiv*, (2020)
5. X. Ouyang, J. Huo, L. Xia, F. Shan, J. Liu, Z. Mo, F. Shi, Dual-sampling attention network for diagnosis of COVID-19 from community acquired pneumonia. *IEEE Transactions on Medical Imaging* (2020)
6. C. Raju, E. Philippsy, S. Chacko, L.P. Suresh, S.D. Rajan, (March), A survey on predicting heart disease using data mining techniques, in *Proceedings of the 2018 Conference on Emerging Devices and Smart Systems (ICEDS)* (2018)
7. S.R. Sonu, V. Prakash, R. Ranjan, K. Saritha, (August), Prediction of Parkinson's disease using data mining, in *Proceedings of the 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)* (pp. 1082–1085). IEEE.S (pp. 253–255). IEEE (2017)
8. , M.Z. Tay, C.M. Poh, L. Rénia, P.A. MacAry, L.F. Ng, The trinity of COVID-19: immunity, inflammation and intervention. *Nat. Rev. Immunol.* **11** (2020)
9. Intel. 2020. Logitboost Classifier. [online] Available at: <https://software.intel.com/content/www/us/en/develop/documentation/daal-programming-guide/top/algorithms/training-and-prediction/classification/boosting/logitboost-classifier.html> [Accessed 17 October 2020]
10. (1, 2, 3) Jerome Friedman, Trevor Hastie, Robert Tibshirani, Additive logistic regression: a statistical view of boosting. *Annals Stat.* **28**(2), 337–374. JSTOR. Project Euclid (2000)
11. Medium. 2020. Understanding Random Forest. [online] Available at: <https://towardsdatascience.com/understanding-random-forest-58381e0602d2> [Accessed 17 October 2020].
12. D.L. Dawson, N. Golijani-Moghaddam, COVID-19: psychological flexibility, coping, mental health, and wellbeing in the UK during the pandemic. *J. Contextual Behav. Sci.* **17**, 126–134 (2020)
13. J.M. Groarke, E. Berry, L. Graham-Wisener, P.E. McKenna-Plumley, E. McGlinchey, C. Armour, Loneliness in the UK during the COVID-19 pandemic: cross-sectional results from the COVID-19 Psychological Wellbeing Study. *PloS One*, **15**(9), e0239698 (2020)

Evolutionary Algorithms for Face Recognition with Mask



Ekansh Chauhan, Manpreet Sirswal, Richa Singh, Nikhil Bagla, Bhaskar Kapoor, and Deepak Gupta

Abstract Due to the Covid-19 pandemic, wearing masks in public places has become a necessity. But it also comes with its challenges, existing face recognition systems are trained to recognize faces with all the features and therefore are failing to work efficiently due to masks. To provide a potential solution to this problem and to recognize faces with masks two evolutionary algorithms, Crow Search Algorithm (CSA) and Cuttle Fish Algorithm (CFA), are used for feature selection which select an optimal subset of features from the existing dataset with vast number of features. In the last step four machine learning classifiers (Support Vector Machine, Random Forest classifier, K-Nearest Neighbor, and Decision tree classifier) are practiced on each subset of features received by both the feature selection algorithms. Experimental results show that CSA removed most of the irrelevant features by selecting only 41% of the original featured and CFA selected 60% of the features. Highest accuracy of classification was received by CSA of 86.5% with Random Forest classifier. Therefore, it shows that CSA and CFA can be used in various other real time applications due to their reduced computational cost and high accuracy.

Keywords Covid-19 · Face detection · Evolutionary algorithm · Crow-search · Cuttlefish

1 Introduction

Due to Covid-19 pandemic people are wearing face masks all over the world. As proven by WHO [1] it helps in impeding the spread of virus, but however wearing masks has created some problems too. Face recognition techniques have failed terribly. It has created a problem of face identification and recognition, and hence has hindered all its applications such as face attendance, security investigation etc. Our

E. Chauhan (✉) · M. Sirswal · R. Singh · N. Bagla · B. Kapoor · D. Gupta
Maharaja Agrasen Institute of Technology, Delhi, India

D. Gupta
e-mail: deepakgupta@mait.ac.in

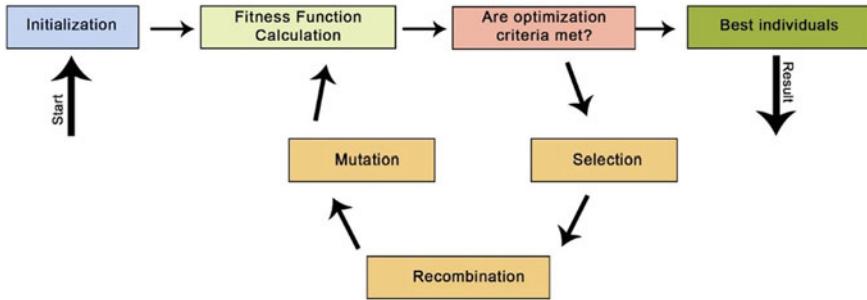


Fig. 1 Process cycle of Genetic Algorithms

security checks rely heavily on face recognition. Public security checks like railway stations rely heavily on traditional face recognition systems but due to face masks they are not operating effectively [2] and removing masks can increase the risk of infection as covid-19 spreads through contact. This also makes other identification systems like finger print also unsafe to use. As face recognition works without any touch, it is the safest unlocking system. To solve the above-mentioned problems, it is necessary to improve existing Face recognition approach that relies on all facial features [3].

Therefore, in this paper an approach to recognize faces with masks is proposed. Evolutionary algorithms are used for feature selection. Evolutionary algorithms are bio-inspired algorithms and they are an active of research for solving different types of optimization problems. Evolutionary algorithms are based on natural evolution of animals. It consists of some main processes namely Initialization, evaluate fitness, selection and reproduction, etc. as shown in Fig. 1 [4].

Using the above ideas we have used two evolutionary algorithms for feature selection, Crow search algorithm (CSA) and Cuttle fish algorithm (CFA). Using a filter-based approach, these algorithms will select optimal subsets of features from a large set of dataset.

The major highlights of the paper are:

1. The prime objective of this study is to recognize faces with mask.
2. Two evolutionary algorithms are implemented, Crow search algorithm (CSA) and Cuttle fish algorithm (CFA).
3. The two evolutionary algorithms CSA and CFA are used to select relevant features from the pool of extracted image features.
4. First subset of features is created using data augmentation and Feature extraction on the image dataset.
5. Classification is performed using four different classification models. The rest of the application is as follows: Sect. 2 presents the methodology used along with the two feature selection methods and their implementations in detail. Section 3 discusses the results. Section 4 finally concludes the paper.

2 Methodology

2.1 Dataset

The dataset was taken from a GitHub repository called “X-zhangyang/” [5], it consisted of 405 masked images of different people. The reason for using this source for dataset is that these sources contain images of very diverse people from different countries which is very important for a face recognition system to recognize faces from around the world. All the images from this source are available to the general public and researchers too. The methodology of this study is shown in Fig. 2.

3 Data Augmentation

To increase the size of our dataset from 405 to 2806 images, several data augmentation techniques like flip, rotation, scale, crop and translation were used. Data augmentation makes minor alterations in our existing dataset and increases the diversity in our dataset by applying random transformations [5]. The classifier will take those images as distinct images and hence will increase the meaningful data. In our dataset, data augmentation was done using Keras pre-processing layers [6]. A sample of before and after images of data augmentation are given in Fig. 3.

4 Feature Extraction and Normalization

Feature extraction is the process of converting image dataset to comma separated values (csv) file. Feature extraction derives values (features) from the existing image dataset in terms of pixels or texture. We used PIL library in python for feature extraction. During feature extraction of 2806 images, 65,536 features were derived.

After feature extraction, data normalization was performed. In data normalization, the data is organized and converted to be in coherence with the values of other attributes or features, which increases the consistency and accuracy of the model. Each feature is scaled within a given range. Sklearn library was used to perform data normalization. Since MinMaxScaler was used the feature range was (0,1) i.e. the values of all features were translated between 0 and 1.

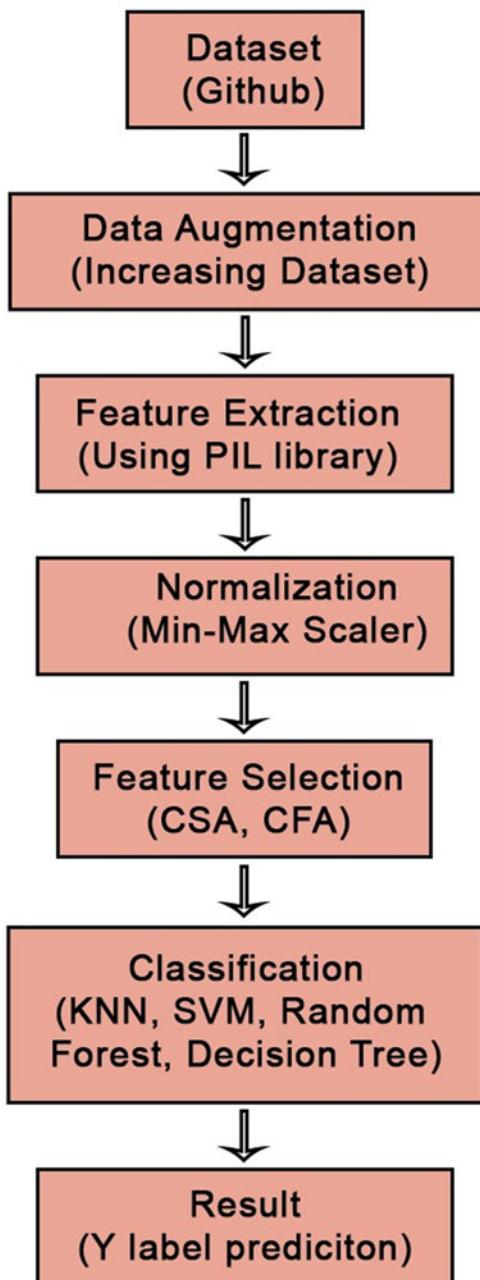
Fig. 2 Process Flow chart



Fig. 3 Sample images of Dataset

5 Feature Selection

Feature selection is the process of removing noise from data, i.e. the features that contribute a negligible amount to the output or are irrelevant and redundant. Therefore, it is process of selection a subset of data from the whole dataset, which contains all the required features. This is done to reduce computational cost and increase the accuracy as having irrelevant or redundant features in the dataset can decrease the accuracy of the model. In this project Crow search algorithm [7] and Cuttle fish algorithm [8] are used for feature selection. They both are evolutionary algorithms and hence are inspired from natural evolution of animals.

5.1 *Crow Search Algorithm (CSA)*

“Askarzadeh proposed an algorithm called Crow Search Algorithm (CSA) (Fig. 5), a metaheuristic bio-inspired optimizer inspired by the intelligence of crows. Crows are considered the most intelligent birds worldwide, they are famous for their intelligence and sharp memory. Studies have proven that crows are capable of remembering faces”. They are famous for snooping on other birds and learning their food hiding places, and then stealing that food when the owner birds are not around. This is the prime idea behind CSA. Since crows are insatiable in nature, they attempt to take each other’s food as well. What’s more, to keep their food from being taken they utilize probabilistic approach. For an optimization problem “crows are considered as search agents, the environment is assumed to be the search space, each position of the environment is considered to be a feasible solution, fitness function defines the quality of the food source and the global solution of the optimization problem is the best food source in the environment”. Dependence of the algorithm on the flight length is shown in Fig. 4 [9] and all the parameters used are given in Table 1.

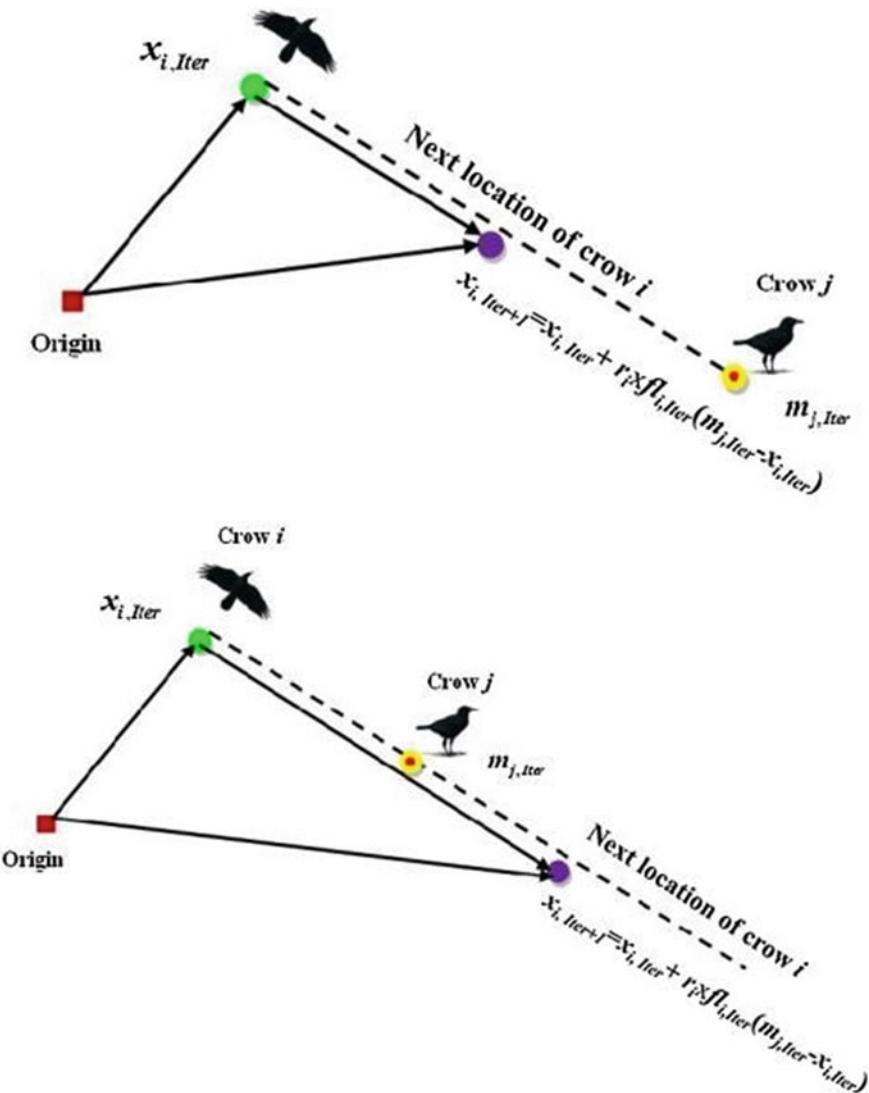


Fig. 4 Dependence of CSA on flight length. Crow i can go to every position on the dash line

5.2 Cuttle Fish Algorithm (CFA)

According to [8], “In 2013, Eesa, Brifcani, Orman proposed a meta-heuristic bio-inspired optimization algorithm called Cuttlefish Algorithm (CFA) (Fig. 7) to solve the numerical global optimization problems”. CFA is based on the color changing characteristics of cuttle fish. The concept behind cuttle fish algorithm is that using reflection of light, cuttle fish creates mesmerizing patterns using chromatophores,

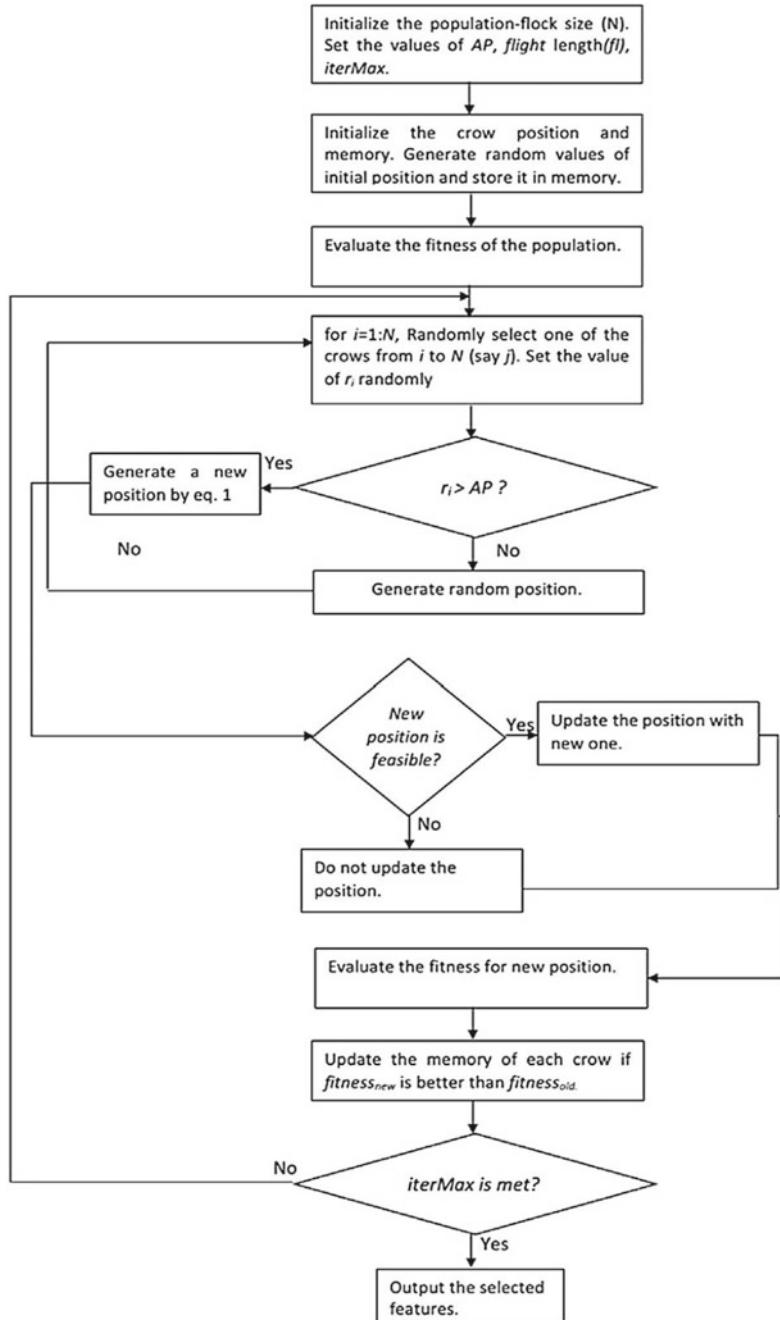
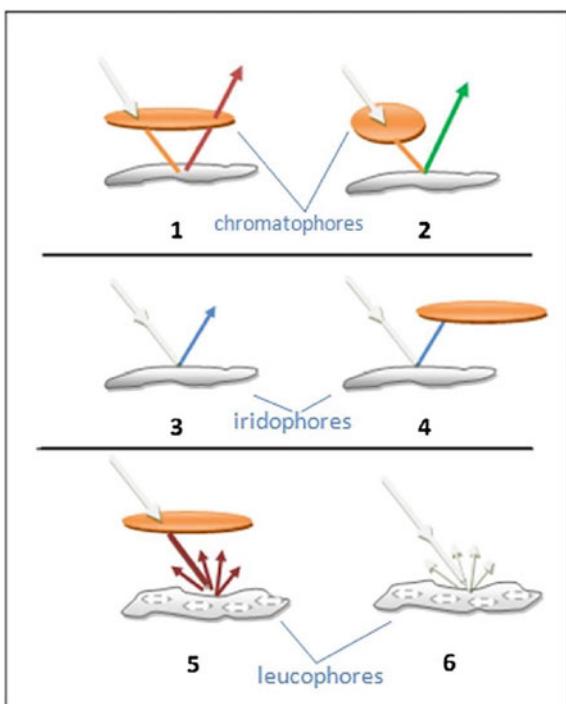
**Fig. 5** Crow search algorithm

Table 1 Parameters used in Crow search algorithm

Parameters	Value
Total crows	20
Awareness Probability	0.1
Iterations	200
Total features	65,536
Weight Factor)	1.6
Flight length	0.2

leucophores and iridophores (Fig. 6). These three are the layers of cells that a cuttle fish possesses. The amalgamation of these three layers forms the six different possibilities of reflection of light. All these 6 cases are split into four groups, called G1, G2, G3 and G4. All of these groups are independent of each other. All the parameters used are shown in Table 2.

Fig. 6 Six distinct cases of light reflection through cell layers



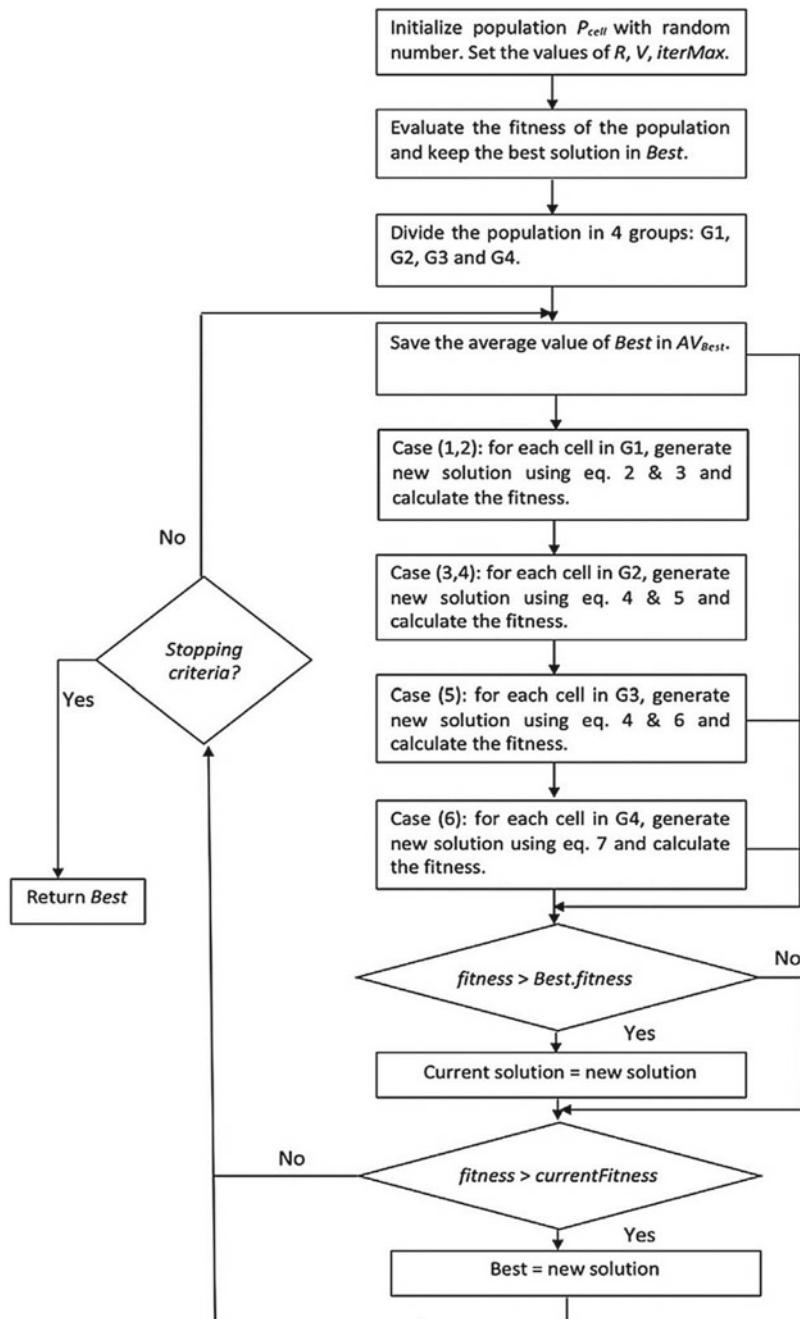


Fig. 7 Crow search algorithm

Table 2 Parameters used in Cuttle Fish algorithm

Parameters	Value
Cell Population	20
No. of Dimensions	0.1
Group Size	10
Iterations	65
Visibility degree used in case 1 & 2	1
Reflection degree used in case 3 & 4	1
Reflection degree used in case 5	1
Lower bound of initial weights	-1
Upper bound of initial weights	1

5.3 Classification

For the classification of images, four different classifiers are used namely, Random Forest Classifier, Support Vector Machine (SVM), K-Nearest Neighbours (k-NN) and Decision Tree.

Classifier.

5.3.1 Random Forest Classifier

According to [10], “Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. Random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting”. The major parameters used for random forest are n_estimators = 50, sample_split = 2, bootstrap = 2. All the other parameters are shown in Table 3.

Table 3 Parameters used in classifiers

Model	Tuning parameters
KNN	n_neighbors = 6, metric = ‘minkowski’
SVM	kernel = linear, gamma = 1, C = 1.0, epsilon = 1, shrinking = true
Random Forest	n_estimators = 500
Decision Tree	criterion = ‘gini’, splitter = ‘best’, max_depth = 10

5.3.2 Support Vector Machines (SVM)

It is a supervised Machine Learning model which is used for classification problems. It works by creating a hyperplane or a set of hyperplanes in an infinite dimensional space [11]. The parameters used for support vector machine are: Kernel = linear, gamma = 1, C = 1.0, epsilon = 1. They are also shown in Table 3.

5.3.3 K-Nearest Neighbors (KNN):

According to [12] “ k -NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until function evaluation. Since this algorithm relies on distance for classification, normalizing the training data can improve its accuracy dramatically”. The parameters used for K-NN are n_neighbors = 6, weights = “uniform”, metric = “minkowski”.

5.3.4 Decision Tree Classifier:

Decision tree is a classification technique that uses a very straightforward idea. It organizes a series of test questions and conditions in a tree structure [13]. The parameters used for Decision Tree Classifiers are max_depth = 10, min_samples_split = 20 and all other default parameters are shown in Table 3.

6 Results

For Feature selection, crow search algorithm (CSA) gave the best results, it selected only 26,869 features from the set of 65,536 features, i.e. 41% (Fig. 8) and Cuttlefish Algorithm (CFA) selected 39,321 features from 65,536 features i.e. 60% (Fig. 9) and comparison bar plot has been generated to clearly observe the difference in the number of features selected by both the algorithms (Fig. 10). Based on that, clearly CSA removed more irrelevant features, as shown in Table 4.

When the data extracted from CSA was put into different classifiers the accuracy was as follows KNN = 84.6%, Random Forest = 86.5%, SVM = 83.2% and Decision tree = 81.8%.

While the data extracted from CFA was put into different classifiers the accuracy was as follows KNN = 81.4%, Random Forest = 82.4%, SVM = 78.2% and Decision tree = 76.1% as shown in Table 5 and Fig. 11.

So, out of CFA and CSA, CSA performed more efficiently. And out of the four classifiers used Random Forest gave the best accuracy.

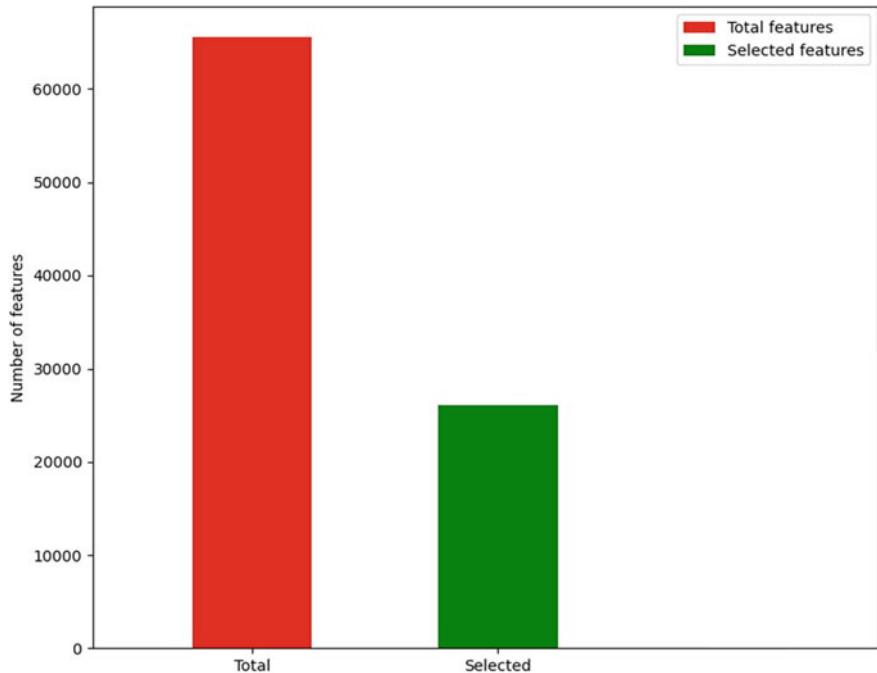


Fig. 8 Bar graph of number of features selected by CSA

7 Conclusion and Future Scope

In the presented work, two evolutionary algorithms were used on the masked face dataset, for feature selection, to improve the accuracy of classification and to reduce the computational cost. After that four machine learning classifiers were applied to each subset of features obtained using the feature selection algorithms. It was observed that Crow Search algorithm gave the best accuracy with Random Forest classifier followed by KNN. Subsequently high level of accuracy was received by both the feature selection algorithms. The two evolutionary algorithms can be used in various other fields. For example, it can be implemented to identify and classify diseases in medical science such as chest CT scan images.

Several other bio-inspired algorithms can be explored for feature selection as a future work. The proposed algorithms can be applied in a wide range of research areas having global optimization problems. This work may be extended by considering more evolutionary algorithms. They can be used as feature selection methods by combining with other classification models and deep learning techniques for obtaining more accurate results with lesser computational times.

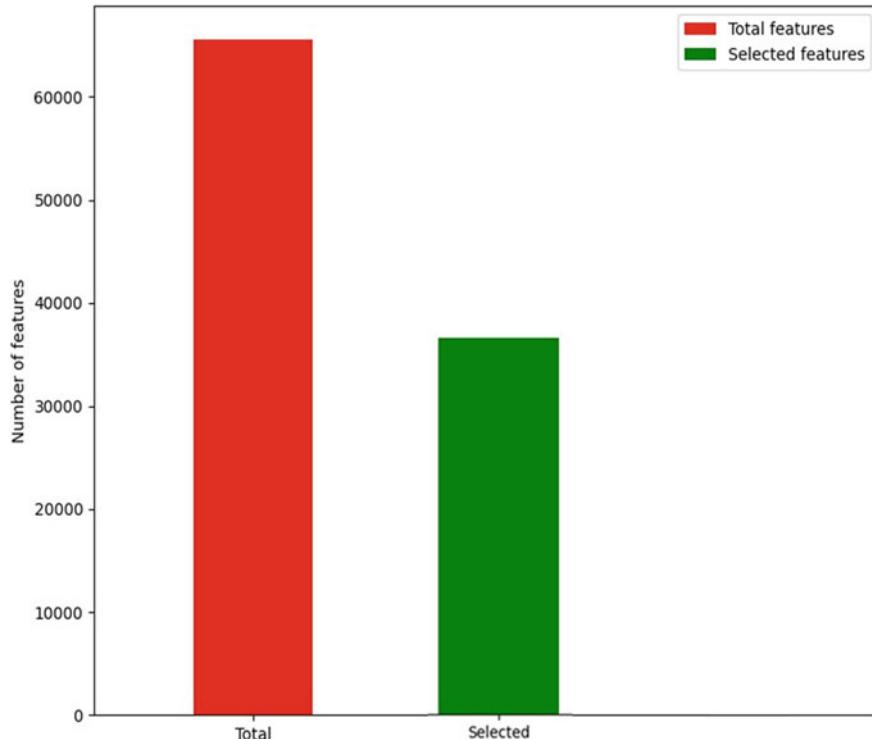


Fig. 9 Bar graph of number of features selected by CFA

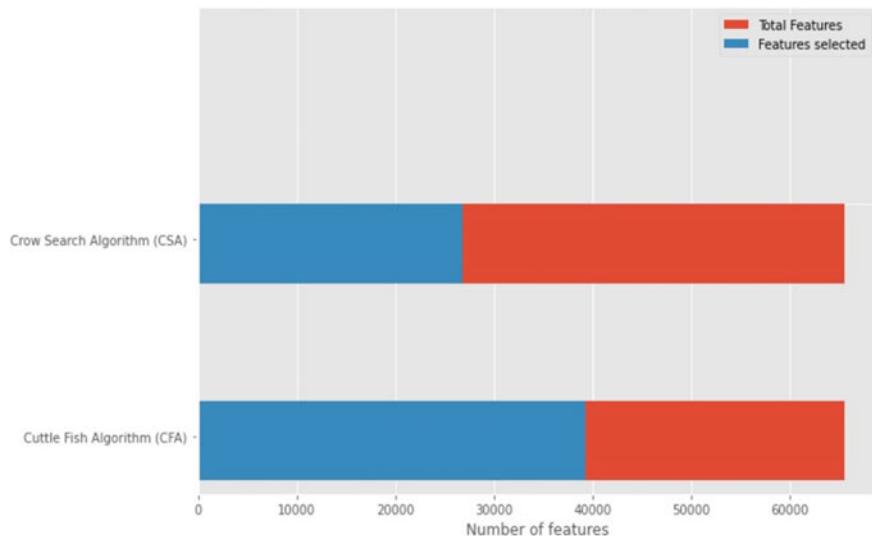


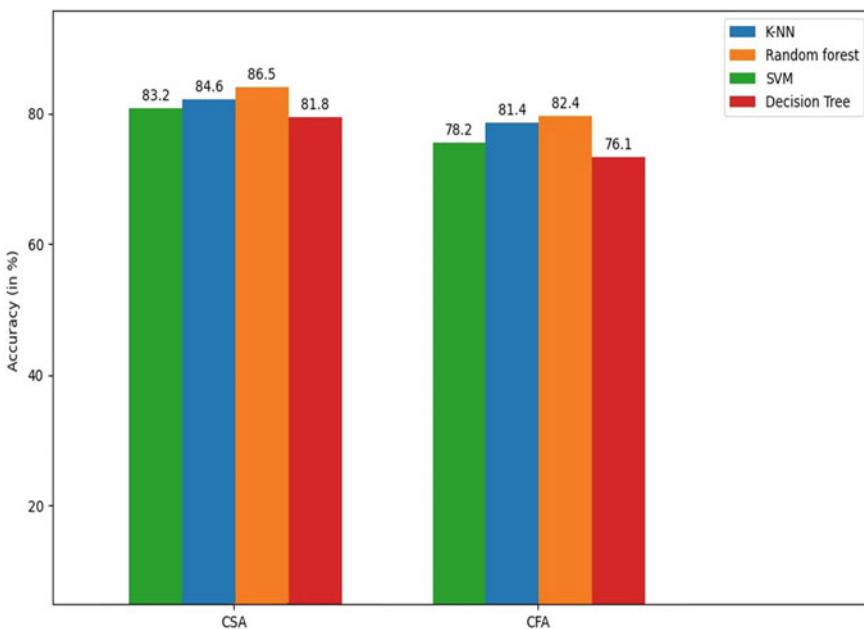
Fig. 10 Comparison between number of features selected by CSA and CFA

Table 4 Features selected by evolutionary algorithms

Feature selection method	No. of features selected	Total number of features
Crow Search Algorithm	26,869 (approx. 41%)	65,536
Cuttle Fish Algorithm	39,321 (approx. 60%)	65,536

Table 5 Accuracy of model for each classifier

Method	Classifier	Accuracy (%)
Crow Search Algorithm (CSA)	k-NN	84.6
	Random Forest	86.5
	SVM (Linear)	83.2
	Decision Tree	81.8
Cuttle Fish Algorithm (CFA)	k-NN	81.4
	Random Forest	82.4
	SVM (Linear)	78.2
	Decision Tree	76.1

**Fig. 11** Comparison between accuracy obtained from different classifiers for two evolutionary algorithms

References

1. World Health Organization (WHO), “Novel Coronavirus–China,” *World Health Organization, disease outbreak news.* (2020)
2. B. Rai, A. Shukla, and L. K. Dwivedi, “COVID-19 in India: predictions, reproduction number and public health preparedness,” *medRxiv.* (2020), doi: <https://doi.org/10.1101/2020.04.09.20059261>
3. D. Comaniciu, “Artificial intelligence for healthcare”. (2020), doi: <https://doi.org/10.1145/3394486.3409551>
4. T. Bartz-Beielstein, J. Branke, J. Mehnen, and O. Mersmann, “Evolutionary algorithms”, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery.* (2014), doi: <https://doi.org/10.1002/widm.1124>
5. Z. Wang et al., “Masked face recognition dataset and application,” *arXiv.* (2020)
6. C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning. *J. Big Data* (2019). <https://doi.org/10.1186/s40537-019-0197-0>
7. Keras, “Image preprocessing—Keras documentation,” *Keras Documentation,* (2019)
8. A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. *Comput. Struct.* (2016). <https://doi.org/10.1016/j.compstruc.2016.03.001>
9. A. Sabry Eesa, A. Mohsin, A. Brifcani, and Z. Orman, “Cuttlefish algorithm—a novel bio-inspired optimization algorithm,” *Int. J. Sci. Eng. Res.*, (2013)
10. N. Gupta, D. Gupta, A. Khanna, P. P. Rebouças Filho, and V. H. C. de Albuquerque, “Evolutionary algorithms for automatic lung disease detection”, *Meas. J. Int. Meas. Confed.*, (2019), doi: <https://doi.org/10.1016/j.measurement.2019.02.042>
11. Y. Qi, “Random forest for bioinformatics”, in *Ensemble Machine Learning: Methods and Applications,* (2012)
12. R.G. Brereton, G.R. Lloyd, Support vector machines for classification and regression. *Analyst.* (2010). <https://doi.org/10.1039/b918972f>
13. Z. Zhang, Introduction to machine learning: K-nearest neighbors. *Ann. Transl. Med.* (2016). <https://doi.org/10.21037/atm.2016.03.37>
14. P.E. Utgoff, N.C. Berkman, J.A. Clouse, Decision tree induction based on efficient tree restructuring. *Mach. Learn.* (1997). <https://doi.org/10.1023/A:1007413323501>

Stock Price Prediction Using Reinforcement Learning



Poonam Rani, Jyoti Shokeen, Anshul Singh, Anmol Singh, Sharlin Kumar, and Naman Raghuvanshi

Abstract With the availability of new data sources and advancement in marketing and financial instruments, stock market returns are a major research area. Stocks have a huge influence on today's economy. A better predictive model is extremely important in stock prediction. The aim of this paper is to investigate the positive effect of reinforcement learning on stock price prediction techniques. Q-learning has been shown to be incredibly effective in various segments, such as cloud scheduling and game automation. This paper demonstrates how the Q-learning technique is helpful in stock price prediction. The findings are very positive with excellent predictive accuracy and meteoric speed.

Keywords Reinforcement Learning · Stock Price · Stock Market Prediction · Q-learning

P. Rani (✉) · A. Singh · A. Singh · S. Kumar · N. Raghuvanshi

Department of Computer Engineering, Netaji Subhas University of Technology, Dwarka New Delhi, India

e-mail: anshuls1.co.17@nsit.net.in

A. Singh

e-mail: anmols.co.17@nsit.net.in

S. Kumar

e-mail: sharlink.co.17@nsit.net.in

N. Raghuvanshi

e-mail: namanr.co.17@nsit.net.in

J. Shokeen

Department of Computer Science and Engineering UIET, Maharshi Dayanand University, Rohtak, Haryana, India

e-mail: jyotishokeen.rs.uiet@mdurohtak.ac.in

1 Introduction

Today, stock exchanges are one of the most profitable fields of stock trading in the business world. As the industry is becoming prevalent, there is an overwhelming need for better and faster prediction models as stocks directly impact the future of the business as well as the future of investors. The stock exchange can be thought of as a game where you have to sell, buy, and hold stocks at the right time by analyzing the financial market and in some cases with your gut. But still, no one can predict the future which makes this game extremely hard and risky. How accurately you can predict the future of the company will decide your victory, and in this case, victory could mean millions of dollars earned in no time, but at the same time loss would mean millions of dollars lost in no time. This is why there is an escalating need for an efficient stock price prediction model. Consequently, stock prediction is a hot topic in the research and development sector.

Many stock prediction models, including Convolutional Neural Network (CNN), Recurrent Neural Networks (RNN), and Long short-term memory (LSTM) are based on neural networks [2]. Reinforcement learning is one of the machine learning techniques that deals with how the agent performs actions to maximize cumulative reward in an environment [11]. This technique functions on the reward and punishment policy. It implies that the model penalizes each time it does not work for the solution, and gives reward if the action turns into victory. Reinforcement learning offers simple solutions to several complex problems than other supervised and unsupervised machine learning algorithms.

In the past decade, stock traders had to rely on different software intelligence systems to reach trading decisions. Lately, with the evolution of artificial intelligence networks, this field has completely changed and has experienced a huge reform. Apart from reinforcement learning, there are numerous machine learning algorithms that can be used efficiently in stock prediction such as CNNs, RNNs, and LSTM with sliding windows. We employ reinforcement learning for stock price prediction in this paper because reinforcement learning allows the use of market signals to create profitable trading strategies in a trading context.

The subsequent sections are organized as follows: Sect. 2 introduces some recent works related to this area in the literature. Sections 3 and 4 introduce the reinforcement learning approach and stock price prediction, respectively. Section 5 defines the methodology used in the paper. Section 6 discusses the experimental work and the results. Lastly, Sect. 7 concludes the paper.

2 Related Works

Parmar et al. [4] also worked in this direction of stock market prediction to predict future values of financial stocks. They used linear regression and LSTM to propose the model. Linear regression is used for predicting continuous values by reducing

the error function, i.e., gradient descent. They used LSTM for prediction on a large amount of data. However, LSTM needs a huge amount of historical data for training purposes to get good accuracy. Compared to Q-learning, LSTM requires more memory for the training dataset. Also, Q-learning provides a sense of random actions to be taken like humans, which is not possible in LSTMs. LSTMs can just predict stock prices but cannot take actions like buy, sell, or hold according to the predictions.

In paper [7], the authors used Support Vector Machine (SVM) as the classifier to know about the action to perform. They believe that SVM is one of the most suitable algorithms for time series prediction. Li et al. [3] employed deep reinforcement learning for stock transaction strategy. They claimed that their algorithm is more intelligent than traditional algorithms because of fast adaption and response for changes. However, their approach is not feasible for large datasets.

3 Reinforcement Learning

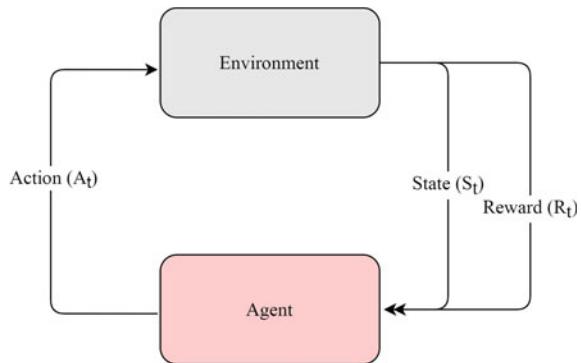
Reinforcement learning is one of the machine learning areas which is concerned with how the algorithms decide to take actions in an environment to augment the cumulative reward. Unlike supervised algorithms, reinforcement learning does not need a labeled input/output dataset. Instead, it focuses on finding the balance between exploration and exploitation. Reinforcement learning works similar to the human brain. We take action based on some past experience and our intuition. We assess the result and reward ourselves if it turns out to be profitable and learn that this is a viable action. But in case of loss, we penalize ourselves and try a different way to solve the problem. This is how reinforcement learning works: it grants rewards if the algorithm's action results in a win and penalizes if it loses. It learns each time it makes a prediction. Figure 1 portrays the functioning of reinforcement learning.

Q-learning is an off-policy reinforcement learning approach that aims to get the best action from the given current state. Q-learning is treated as off-policy because it does not need any policy and the Q-learning function learns from actions that are outside the current policy such as taking a random action. “Q” stands for quality in Q-learning. Quality indicates how productive a given action is in gaining some future reward. The goal of Q-learning is to maximize the total reward [1]. Li et al. [3] applied deep reinforcement learning in stock forecasting.

4 Stock Price Prediction

In the financial world, forecasting stock prices is an essential goal. A fairly accurate forecast has the potential to yield high financial benefits and protect against market risks. An efficient stock price prediction system can result in a huge amount of profit in the future. The theory of an efficient market implies that stock prices represent all the information currently available, and any price adjustments that are not based on newly

Fig. 1 Reinforcement learning



released information are therefore potentially unpredictable. Others disagree and those with this perspective have countless techniques and technologies that allegedly allow them to gain information on future prices. However, due to the uncertainty and unpredictable nature of the markets and the many undecidable, non-stationary stochastic variables involved, forecasting stock prices is not a simple task. Nowadays, social network analysis is useful in predicting stock prices [6, 8]. On predicting the stock prices, the users can use the recommender systems to buy or sell the stocks [5, 9, 10].

The historical trends of financial time series have been analyzed by several scholars from different areas, and different methods for forecasting stock prices have been proposed. Most of these methods involve careful selection of input variables to achieve promising results, developing a predictive model with skilled financial expertise, and introducing different statistical methods for arbitrage analysis, making it impossible for individuals outside the financial sector to use these methods to forecast stock prices.

5 Methodology

The method that we used in this paper for stock price prediction is Q-learning. We are here first creating an environment and an agent. The term environment in reinforcement learning is referred to as the task, i.e., stock price prediction and the agent refers to the algorithm used to solve that particular task. Hence, the driver program just initiates the needed environment and agents which are given as input to the algorithms which return predictions in values. This part of the algorithm is responsible to calculate the gradient descent or the algorithm which eventually talks about the accuracy of the algorithm.

We incorporate two additional functions, i.e., the reset function and the step function. The reset function's task is to bring back the pointer to zero, i.e., start of the time, where the cash in hand is maximum and the investment is zero. The step function

takes in action as the input and performs action accordingly, i.e., it buys the stock, moves the pointer, and at the same time updates the reward, next state, and portfolio values.

Unlike the typical $Q(s, a)$, we use only the state s and ignore the action a in this stock problem.

$$Q(s, :) = W^T \cdot s + b \quad (1)$$

where $Q(s, :)$ is the vector of Q values at state s , W is the matrix of weights, s is the state, and b is bias.

This part of the algorithm contains mainly three functions with various tasks, i.e., *_init_* function, *get_action* function, and the *train* function. The *_init_* function is just used to initialize the model used for training purposes. The *get_action* function takes the state as the input and accordingly decides which action to be taken, i.e., whether to buy stock, sell stocks, or remain sprawl using reinforcement learning techniques such as epsilon or greedy. Finally, the *train* function takes a tuple of data including current state, action, reward, next state, and done flag. It calculates the input and target values that are input to our model, where input is the state and the target is calculated as follows:

$$\text{target} = r + \gamma * \max Q(s', :) \quad (2)$$

where γ is the discount factor which is used to align existing benefits and potential ones. $\max Q(s', :)$ is the maximum Q value among all possible actions given state s' .

6 Experimental Work

In order to make it a real-time project, it is necessary to take recent data. API Alpha Vantage is used to get real-time data to make good predictions about a stock. Alpha vantage is a free API which is used to provide real-time data of the stock. We take the stock prices of three companies: Apple, Microsoft, and International Business Machines (IBM) for the duration of January 2012–November 2020. The dataset is divided into a 60:40 ratio as training set and testing set, respectively.

The first step to every reinforcement technique is to decide on an action in the beginning. It is assumed that before making any decision, the algorithm must know the answers to some questions such as

- Do I even have enough cash to buy?
- Considering the current state of my portfolio and the existing price of the shares in the market, is it worth selling them?

After answering all these questions, the next step is to decide the action to be taken. There are three actions: buy, sell, or hold. Reinforcement learning algorithms evaluate the action and make the next decision accordingly. Reward is the difference

Table 1 Initial model configurations

Parameter	Value
Episodes	200
Initial investment	20000
ε	1.0

Table 2 Parameters chosen for experiments

Parameter	Value
Exploration rate	1.0
Epsilon decay	0.996
Discount factor	0.94

Table 3 Performance results

Performance parameter	Reward
Average reward	38356.61
Minimum reward	23091.67
Maximum reward	54582.06

between portfolio values of recent time steps and previous time steps. The algorithm computes the portfolio value as follows:

$$\text{value} = S^T \cdot P + C \quad (3)$$

where S is the vector of shares owned, P is the vector of share prices, and C is the cash.

Epsilon decay is the value of ε to learn and act optimally in the life of an agent. We experimented on different values of epsilon_decay and γ to find their best value. Table 1 depicts the initial configurations set in the proposed model.

Table 2 defines the parameters tuned for the experiments and Table 3 shows the performance of the model based on the selected parameters. Figure 2 depicts the results of rewards in respect of epsilon_decay for $\gamma = 0.95$, $\varepsilon = 1.0$, and $\varepsilon_{\min} = 0.01$. The average reward is 32697.13 and the best value for epsilon_decay in terms of reward is 0.996. Figure 3 depicts the results of rewards in respect of γ with exploration rate $\varepsilon = 1.0$, $\varepsilon_{\min} = 0.01$, and $\varepsilon_{\text{decay}} = 0.996$. The average reward is 81588.52 and the best value for γ in terms of reward is 0.94.

Fig. 2 Test results of rewards versus ϵ_{decay}

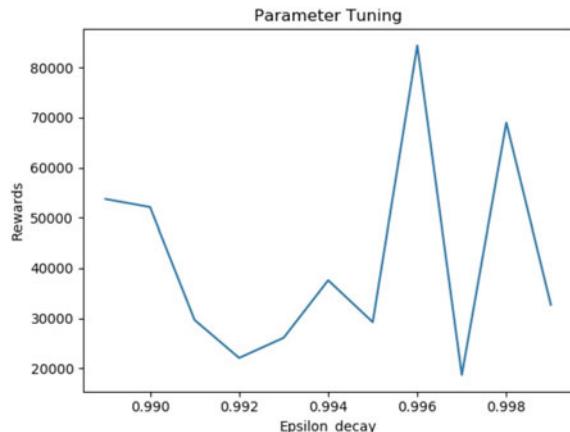
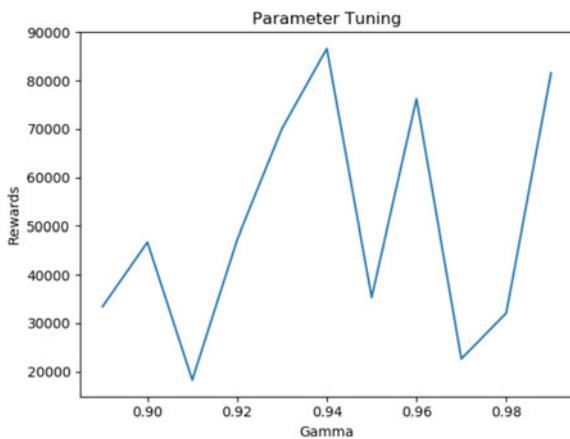


Fig. 3 Test results of rewards versus γ



7 Conclusion

In this paper, we used reinforcement learning for stock price prediction. Also, we found its performance which allows us to use this model as a base of further study in this domain. Reinforcement learning is better than other learning algorithms as it learns from current situations and also makes faster adaptive changes.

References

1. C. Jin, Z. Allen-Zhu, S. Bubeck, M.I. Jordan, Is Q-learning provably efficient? in *Advances in neural information processing systems* (2018), pp. 4863–4873
2. C.K.S. Leung, R.K. MacKinnon, Y. Wang, A machine learning approach for stock price prediction, in *Proceedings of the 18th International Database Engineering & Applications Symposium* (2014), pp. 274–277
3. Y. Li, P. Ni, V. Chang, Application of deep reinforcement learning in stock trading strategies and stock forecasting. *Computing* 1–18 (2019)
4. I. Parmar, N. Agarwal, S. Saxena, R. Arora, S. Gupta, H. Dhiman, L. Chouhan, Stock market prediction using machine learning, in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)* (IEEE, 2018), pp. 574–576
5. P. Rani, J. Shokeen, D. Mullick, Recommendations using modified k-means clustering and voting theory. *Int. J. Comput. Sci. Mobile Comput.* **6**(6), 143–148 (2017)
6. P. Rani, D.K. Tayal, M. Bhatia, SNA using user experience, in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)* (IEEE, 2019), pp. 125–128
7. V.K.S. Reddy, Stock market prediction using machine learning. *Int. Res. J. Eng. Technol.* **5**(10)(2018)
8. J. Shokeen, C. Rana, Social recommender systems: techniques, domains, metrics, datasets and future scope. *J. Intell. Inform. Syst.* **54**, 633–667 (2019). <https://doi.org/10.1007/s10844-019-00578-5>
9. J. Shokeen, C. Rana, A study on features of social recommender systems. *Artif. Intell. Rev.* **53**(2), 965–988 (2020)
10. J. Shokeen, C. Rana, P. Rani, A trust-based approach to extract social relationships for recommendation, in *Data Analytics and Management* (Springer, 2020), pp. 51–58
11. R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction* (MIT press, 2018)

Sentiment Analysis of Bangla Text Using Gated Recurrent Neural Network



Nasif Alvi, Kamrul Hasan Talukder, and Abdul Hasib Uddin

Abstract Sentiment analysis is a fundamental part of Natural Language Processing. There are numerous works on this topic in English and other languages. However, it is still a comparatively new practice in Bangla. The absence of a suitable Bangla corpus is the primary obstacle for sentiment analysis tasks in Bangla. Nonetheless, Long Short-Term Memory (LSTM) is a common technique for resolving sentiments from a dataset containing a large amount of text data. However, Gated Recurrent Unit (GRU) is very efficient for datasets with a low amount of text data. In this manuscript, we present a five-layered GRU neural network model, each layer comprising of 48 neurons, applied the model on an existing Bangla corpus. We have implemented the ten-fold cross-validation approach and repeated the same processes three times. Each time, we have considered the averages of the ten validation accuracy and losses and compared the results with the state-of-the-art published outcome (77.85% highest accuracy) for Bidirectional LSTM (BLSTM). The highest accuracy for our model is 78.41%, while the lowest accuracy is 76.34%.

Keywords Sentiment analysis · Natural language processing · Corpus · Neural network · Text data · LSTM · GRU · BLSTM

1 Introduction

Sentiment Analysis (SA) is a technique to find as well as to classify thoughts expressed in a portion of perusal based on several types of terminologies like computer technology especially in order to decide if the conduct of the writer against a particular subject, upshots, and so on is definitive, opposite or indifferent. Sentiment analysis frequently applies to management among ideas, thoughts as well as temporal texts. SA offers detailed information pertaining to universal judgments since it runs into the entire various forms of prattles, ratings as well as feedback. SA is basically kind of validated tactic for forecasting a variety of important situations, such as box

N. Alvi · K. H. Talukder · A. H. Uddin (✉)
Khulna University, Khulna, Bangladesh

office film reviews as well as universal and provincial particles. Universal views are applied to worth a particular motive like an individual, commodity, or venue, as well as it can be seen on various websites such as Amazon and Yelp. It is possible to define emotions in definitive, opposite, or indifferent classes as well as major tribes. If the conductor has a satisfying and affirmative experience and bad impact, SA will instantly discover the articulate course of user feedback or opinions. In the area of classification of emotion, views or people's emotions are analyzed. In social media and in virtually any system, these kinds of programs are used. The views or emotions are the resemblances of the values, choices, and actions of individuals. With these techniques, it is possible for corporations to make political decisions. In current years, a great number of individuals are sharing their opinions or thoughts through the internet using Bangla [1].

The development of restaurants across numerous online channels can be observed over the last few years. Websites have turned out as the most common forum where restaurants are upheld on the principle of the opinions of customers. The representation of consumer sentiment results from such online customer feedback that magnifies a restaurant's overall quality. The contact between customers and owners through the online portal provides the ability to examine the response of the customer's insights. It is therefore necessary to be able to measure consumer opinion in order to improve quality according to the demands of consumers. The advantage of potential research will be given by a qualified computer by labeled data. Works such as the CNN model for Bangla reviews for factor extraction, mixed machine learning models for forecasting reviews. Sentiment analysis has already been a common form of forecasting consumer ratings. Few research has been performed on the Bangla text in particular, but not so effectively. A Sentiment Analysis model for restaurant ranking was developed by authors on the basis of food price, quality, operation, ambience, and special meaning [2].

SA is basically an implementation of physical dialect technology. It is recognized as concept mining, sentiment elimination. At present, Sentiment Analysis vastly refers to the countable provision of thoughts or analysis, computational linguistics, natural language processing as well as biometrics for the systematic detection, retrieval, quantification, as well as study of affective states and subjective knowledge. Moreover, recent advances in research into machine learning, especially deep research, methods focused on learning, for example, recurrent neural network (RNN), accept advantage of the ability to infer choices through formulating a diagram in SA [3].

Micro-blogging platforms such as Twitter, YouTube, Facebook, etc., have now become very popular for social connections. Through social media, people communicate their sadness, which can be studied to determine the reasons behind their depression. Most studies on the study of emotions as well as depression are focused on inquisitions as well as scholarly interviews in non-Bengali languages, especially English. In identifying human depression, these conventional approaches are not always sufficient. Artificial Intelligence's aim is to mimic human habits, then evaluate them. Machine learning, as well as deep learning approaches, are nowadays being vastly used for analyzing human behavior as well as human sentiment. Detecting

emotion and analyzing sentiment has become an important part and for this several types of learning methods are used. It is possible to further study the classification of feelings and emotions from two separate viewpoints, especially the detection of feelings as well as emotions from image data, and the detection of feelings as well as emotions from textual data. The total field of definitive, opposite as well as mystical emotion alignment works is covered in a common way by sentiment analysis. Emotions, e.g., happiness, grief, depression, disgust, etc., are very profound emotions that are often harder to analyze. Any of those thoughts are stronger than others, needing research of high-level clinical experience as well as much specialized empirical methods. For this reason; sentiment analysis is foremost important [4].

In this research study, we have used Gated Recurrent Unit (GRU) for the sentiment analysis with 7,000 Bangla text data. With the ten-folds cross-validation approach; the process has been implemented and the highest accuracy (78.41%) has been obtained for the sentiment analysis.

2 Related Work

Hoque et al. [3] examined the execution of various ML approaches along with doc2vec for categorizing sentiment of Bangla regular dialects. They streamed a doc2vec model utilizing a corpus developed with seven thousand Bangla sentences and with 120 components of highlight vectors with two kinds of information: positive and negative. Then they utilized a few ML algorithms (LR, SGD, SVM, K-Neighbors Classifier, DT, LDA, SM, BLSTM, and GaussianNB) for analysis where BLSTM acquired the highest accuracy. The information was split 80% as training and the rest 20% as testing haphazardly.

Uddin et al. [5] established a Gated Recurrent Unit model based on depression detection method by analysis. All of the data culled from Bangla information from Twitter, Facebook, and different sources. There were 4 hyper-parameters, specifically, number of GRU layers 5, group size 10, and number of epochs 5. They had collected 5,000 Bangla information from Twitter and 210 depressed Bangla statements from local Bengali speakers utilizing google structure. They utilized GRU size 64, 128, 256, 512, and 1024 for this investigation.

Hossain et al. [6] proposed a joint model with CNN-LSTM to conduct sentiment analysis on online restaurant surveys. They utilized the dataset into 80% for training with CONV size 256 and LSTM size 128. They collected the information of those restaurants that were related to online platform like FoodPanda and Shohoz Food consisting of 1000 reviews Review and category were two sections. At last, the recall, precision, and f1-score average values were 0.70, 0.70, and 0.71.

Sharfuddin et al. [1] accomplished their work on sentiment classification of Bangla content utilizing RNN with BLSTM (Bidirectional LSTM) where contained around 15,000 comments got from Facebook and at that point kept 10,000 comments consisting of 5000 negative comments and 5000 positive comments and all the symbols, emojis, stickers, numbers were erased to work on plain Bangla content.

Hasan et al. [7] developed a model that recognized the sentiment assessment from Bangla text utilizing logical valence examination. This investigation utilized the WorldNet to get the feelings of each word as per its grammatical features (POS) and SentiWordNet to get the earlier valence of each word. That point determined the total positivity, negativity, and neutrality of sentence or archive regarding all-out sense. They made an XML document to store the Bangla word and its related POS and take the assessment of 20–30 people groups about the sentiment of the section.

Tripto et al. [8] introduced an extensive group of methods to recognize sentiment and concentrated on feelings from Bangla texts. In this study, LSTM, SVM, NB, CNN classifiers and the dataset of Bangla sentence along with a 3 class that was affirmative, neutral, negative and a 5 class that was strongly positive, negative, neutral, positive, strongly negative of the estimation name with six fundamental feelings (anger, fear, disgust, sadness, joy, and surprise) were used. They assessed the exhibition of the model utilizing another dataset of Bangla, Romanized Bangla, and English comments from various sorts of YouTube recordings. Their mentioned methods indicated 54.24 and 65.97% accuracy in 3 and 5 names feeling individually.

Al-Amin et al. [9] analyzed a methodology of sentiment characterization and sentiment extraction of words and Bangla comments with word2vec. The dataset had multiline comments and 16,000 Bangla single lines that were gathered from popular blogging websites and tagged every comment to one or the other positive or pessimistic by taking suppositions from various kinds of individuals by overviews. They prepared 90% of the tagged comments picked arbitrarily as well as the leftover 10% because of testing.

In our previous analysis [10] we classified English tweets into five categories: happy, surprise, sad, disgust, and neutral. We used total 4000 tweets as our dataset: 3750 as training set and 250 as test set. Conducting unigram model and unigram using POS tag model 66 and 64.8% accuracies were achieved, respectively.

3 Methodology

The flowchart of the research methodology is shown in Fig. 1.

3.1 Dataset Collection

The dataset was collected from Hoque et al. [3] for sentiment analysis in Bangla text including Positive and Negative sentiment. The total number of samples was 7000 where 3500 samples were positive sentiment and rest of 3500 samples were negative sentiment.

Fig. 1 Working procedure of our system



3.2 Features Extraction

To extract the feature information “Integer encode” method was used in this study. The integer values have a characteristic arranged connection between one another and AI calculations might have the option to comprehend and saddle this relationship. The total length of dataset was obtained 21,889. The total vector size was taken same as the maximum length of sentence. After that, zero padding was used to keep the length of each text same.

3.3 Dataset Training

Ten-fold cross-validation is the most popular technique to train the dataset. It is a re-examining procedure to evaluate predictive models by parceling the first instance into a preparation set to build the model and a test set to evaluate it. It rearranges the dataset haphazardly, sections dataset into 10 set lastly compact the aptitude of the model using the case of model assessment scores. In this study, ten-fold cross-validation (6300 training data and 700 test data) was used and it was calculated 3 times. Each time the training data was shuffled to learn efficiently.

3.4 Gated Recurrent Unit Network (GRU)

GRU network is the streamlined structure of the repetitive neural organization. Notwithstanding, at the point when the info data is expanded to a specific length, the RNN can't associate with the significant data. GRU network is pointed toward tackling the issue of long-range reliance as well as slope vanishing of RNN. The GRU neural organization along with smaller edge structure as well as better productivity is straightforwardly chosen for the determination of stuff pitting shortcoming. Like GRU, an intermittent unit in RNN is recognized as long transient memory. LSTM as well as GRU have the similar objective of following long-haul conditions viably while alleviating the disappearing/detonating inclination issues [11]. The GRU neural organization model adjusts to the issue of reliance on an assortment of time scales by arranging a wide range of cycle units which balance the progression of data with the door unit [12].

In our case, we have used 5 GRU layers with each layer containing 48 neurons. Then the flatten method has been used to convert the whole matrix into one-dimensional vector before the dense layer which is defined as output layer. In dense layer, there remain 2 neurons named positive sentiment and negative sentiment. The activation function has been used “tanh” in hidden layers and, in the final activation, the “softmax” function has been used. For reducing the loss or error rate, the “Categorical_crossentropy_loss” function has been used.

4 Result and Analysis

We have applied ten-fold cross-validation three times on our dataset to compare the results. In each iteration, we have applied shuffling on our training dataset to train the network properly.

Table 1 shows all the validation accuracies and validation losses along with the number of epochs in each fold in the three times running. Figures 2 and 3 represent the graphical view of validation accuracy and validation loss in each fold in all our three

Table 1 Results of tuning GRU Hyper-parameter

Run 1			Run 2			Run 3					
Implementation no	No. of epochs	Validation accuracy (%)	Validation loss	Implementation no	No. of epochs	Validation accuracy (%)	Validation loss	Implementation no	No. of epochs	Validation accuracy (%)	Validation loss
Fold1	9	76.14	0.5033	Fold1	10	73.71	0.5760	Fold1	9	72.14	0.5551
Fold2	11	85.14	0.3924	Fold2	10	84.86	0.3868	Fold2	10	85.71	0.3614
Fold3	13	90.71	0.2626	Fold3	12	89.86	0.2828	Fold3	15	90.00	0.2892
Fold4	11	84.71	0.3270	Fold4	12	85.29	0.3138	Fold4	11	84.86	0.3477
Fold5	10	86.86	0.3193	Fold5	11	87.00	0.3458	Fold5	11	86.29	0.3570
Fold6	10	86.71	0.3429	Fold6	10	84.57	0.3582	Fold6	12	85.57	0.3554
Fold7	10	82.14	0.4216	Fold7	9	81.43	0.4031	Fold7	9	81.14	0.4154
Fold8	7	62.14	0.6760	Fold8	8	63.86	0.6811	Fold8	7	61.57	0.6781
Fold9	13	78.71	0.5899	Fold9	8	71.71	0.5966	Fold9	8	70.00	0.6233
Fold10	6	50.86	0.6971	Fold10	8	58.14	0.6883	Fold10	6	46.14	0.6939

Fig. 2 Validation accuracy of ten-fold cross-validation in three times running

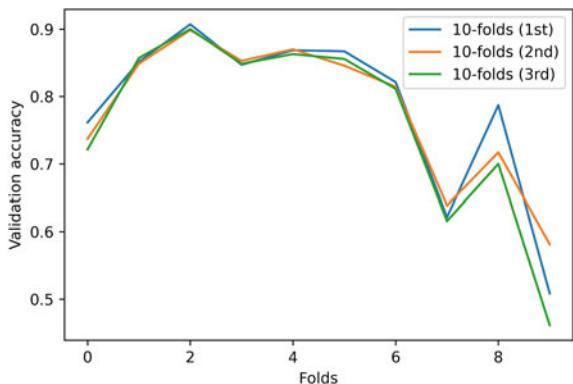
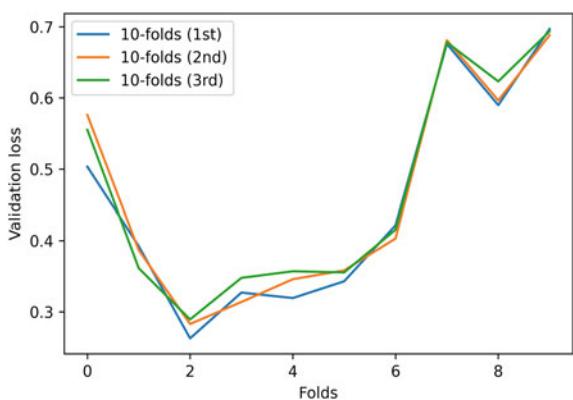


Fig. 3 Validation loss of ten-fold cross-validation in three times running



times running. The lowest validation accuracies were found in fold10 for all three times. On the other hand, we achieved the highest validation accuracy of 90.71% in fold3 in our first iteration using 13 epochs. We achieved average accuracy of 78.41, 78.04, and 76.34% in our three times running, respectively. Figure 4 represents the graphical view of the average accuracy and average loss of all three times running.

Table 2 shows the comparison of our system with Hoque et al. [3]. For analysis, Hoque et al. [3] randomly split their data 80% as training set and 20% as test set. However, randomly splitting a dataset is not the most standard way to learn a model. Because it does not ensure the participation of all data for training. Hence, in our analysis, we applied ten-fold cross-validation three times by shuffling the dataset each time to achieve a more accurate result. Ten-fold cross-validation is one of the most popular and accepted techniques to learn a model while ten-fold cross-validation ensures the participation of all data in training the model. We achieved the superior average accuracy of 78.41% as well as the lowest average accuracy of 76.34%. On the other hand, Hoque et al. [3] achieved the highest accuracy of 77.85% using BLSTM and the lowest accuracy of 59.21% using GaussianNB.

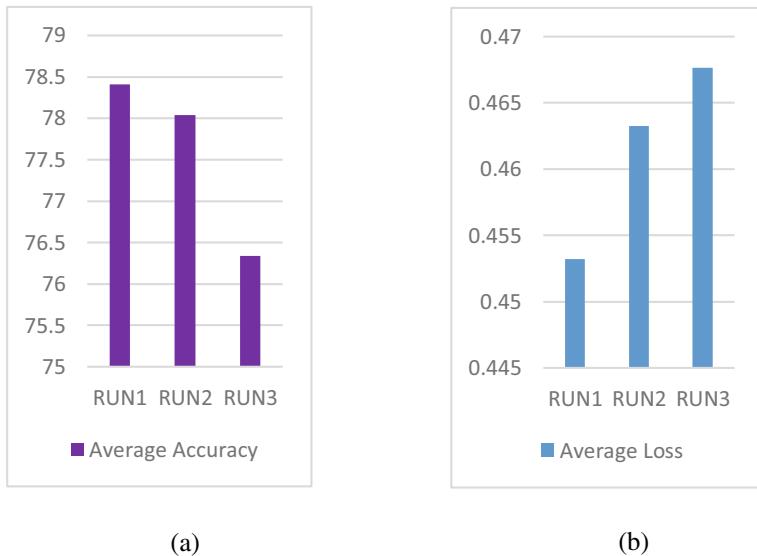


Fig. 4 Graphical view of **a** average accuracy and **b** average loss in three times run

Table 2 Comparison of our system with Hoque et al. [3]

	Our system (%)	Hoque et al. [3] (%)
Highest accuracy	78.41	77.85
Lowest accuracy	76.34	59.21

5 Conclusion

There exist few researches on Bangla text sentiment analysis. For this reason, the dataset on Bangla text is rarely available. In the field of research, sentiment analysis is an emerging topic, we should try to build a more accurate model on native languages. In our research, we have used an existing dataset and we have trained this dataset using GRU. Our system outperforms the previous one. There remain some limitations in our research. In future, we want to apply more preprocessing techniques and other feature extraction methods into our data to get a better result. Also to compare the performances, we want to use other classification algorithms. We want to further continue our study to develop multi-class sentiment analysis or emotion analysis.

Acknowledgements This research work has been funded by Information and Communication Technology (ICT) Division, Ministry of Post, Telecommunication, and Information Technology, Government of the People's Republic of Bangladesh through ICT fellowship.

References

1. A. Aziz Sharfuddin, M. Nafis Tihami, M. Saiful Islam, A deep recurrent neural network with BiLSTM model for sentiment classification, in *Proceeding of the International Conference on Bangla Speech and Language Processing (ICBSLP)*, Sylhet (2018), pp. 14. doi: <https://doi.org/10.1109/ICBSLP.2018.8554396>
2. N. Hossain, M.R. Bhuiyan, Z.N. Tumpa, S.A. Hossain, Sentiment analysis of restaurant reviews using combined CNN-LSTM, in *Proceeding of the 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kharagpur, India (2020), pp. 15. doi: <https://doi.org/10.1109/ICCCNT49239.2020.9225328>
3. M.T. Hoque, A. Islam, E. Ahmed, K.A. Mamun, M.N. Huda, Analyzing performance of different machine learning approaches with Doc2vec for classifying sentiment of Bengali natural language, in *Proceeding of the International Conference on Electrical, Computer and Communication Engineering (ECCE)*, Cox'sBazar, Bangladesh (2019), pp. 1–5. doi: <https://doi.org/10.1109/ECACE.2019.8679272>
4. X. Wang, C. Zhang, Y. Ji, L. Sun, L. Wu, Z. Bao, A depression detection model based on sentiment analysis in micro-blog social network, *Lecture Notes in Computer Science* (2013), pp. 201213. Available: https://doi.org/10.1007/978-3-642-40319-4_18
5. A.H. Uddin, D. Bapery, A.S. Mohammad Arif, Depression analysis of Bangla social media data using gated recurrent neural network, in *Proceeding of the 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, Dhaka, Bangladesh (2019), pp. 16. doi: <https://doi.org/10.1109/ICASERT.2019.8934455>
6. N. Hossain, M.R. Bhuiyan, Z.N. Tumpa, S.A. Hossain, Sentiment analysis of restaurant reviews using combined CNN-LSTM, *ICCCNT*, Kharagpur, India (2020), pp. 15. doi: <https://doi.org/10.1109/ICCCNT49239.2020.9225328>
7. K.M.A. Hasan, Mosiur Rahman, Badiuzzaman, Sentiment detection from Bangla text using contextual valency analysis, in *Proceeding of the 2014 17th International Conference on Computer and Information Technology (ICCIT)*, Dhaka (2014), pp. 292295. doi: <https://doi.org/10.1109/ICCITECHN.2014.7073151>
8. N. Irtiza Tripto, M. Eunus Ali, Detecting multilabel sentiment and emotions from Bangla YouTube comments, *ICBSLP*, Sylhet (2018), pp. 16. doi: <https://doi.org/10.1109/ICBSLP.2018.8554875>
9. M. Al-Amin, M. S. Islam, S. Das Uzzal, Sentiment analysis of Bengali comments with Word2Vec and sentiment information of words, *ECCE*, Cox's Bazar (2017), pp. 186190. doi: <https://doi.org/10.1109/ECACE.2017.7912903>
10. A.Z. Riyad, N. Alvi, K.H. Talukder, Exploring human emotion via Twitter, *ICCCIT*, Dhaka (2017), pp. 15. doi: <https://doi.org/10.1109/ICCITECHN.2017.8281813>
11. X. Li, J. Li, Y. Qu, D. He, Gear pitting fault diagnosis using integrated CNN and GRU network with both vibration and acoustic emission signals. *Appl. Sci.* **9**(4), 768 (2019). Available: <https://doi.org/10.3390/app9040768>
12. B. Liu, C. Fu, A. Bielefield, Y. Liu, Forecasting of Chinese primary energy consumption in 2021 with GRU artificial neural network. *Energies* **10**(10), 1453 (2017). Available: <https://doi.org/10.3390/en10101453>

Leveraging User Comments in Tweets for Rumor Detection



Shaswat Patel, Binil Shah, and Preeti Kaur

Abstract A novel technique is presented in this paper to detect rumors from tweets. Rumor is unverified information at the time of posting. To detect rumors in the tweets, we use transformer models BERT, RoBERTA, ALBERT, and DistilBERT. These techniques perform the feature extractor for input sequence consisting of source tweet and user comments on the source tweet. The key insight is that by understanding the context of the source tweet and the user comments the models can successfully classify the source tweet into rumor or non-rumor. This is based on the fact that users on social media sites try to classify any new information into rumor and non-rumor collectively by using comments. Our approach was able to produce better precision, recall, and F1 score over the state-of-the-art classifier that uses Conditional Random Fields (CRFs) to learn the context during the event.

Keywords Rumor detection · Natural language processing · Deep learning · Transformer models

1 Introduction

Social media has become an integral part of our day-to-day lives. Social media is increasingly used for spreading developments during breaking news stories, especially Twitter. This is thanks to the fact that users of these platforms can share information in real time, enabling users present on the ground to post information as the events unfold in front of them. However, the fast-paced nature of the breaking news and the unmoderated feature of social media platforms combine together to the unable posting of information that is unverified at that time. In the absence of a moderating system for verifying the information on the social media, it has been observed that the social media users will share their thoughts on the truthiness of the information/post using comments in a process of exposing the truth by forming

S. Patel · B. Shah · P. Kaur (✉)

Department of Computer Science and Engineering, Netaji Subhas University of Technology, New Delhi, India

e-mail: preeti.kaur@nsut.ac.in

arguments to either support the information or not [1]. As social media platforms become more and more sought after during a major world event, it is important that some sort of moderating system must be in place to help users distinguish between rumor and non-rumor information. This has led to building a rumor classification system that can act as a filter to reduce the spread of rumors.

We define rumor as “a currently circulating story or report of uncertain or doubtful truth” which is consistent with the Oxford English Dictionary. The unverified rumor may over time become true but at the time of reporting the information is unverified. Unverified means when there is no evidence supporting the information, or the source has no credibility. This definition is consistent with various other works and the dataset [1, 3].

A rumor detection system’s end goal is to warn the users that the information they are viewing is unverified at that point in time, hence letting the users decide the veracity of the information. Rumor detection tasks previously have looked at ‘querying tweets’ or tweets that query the credibility of the information to classify the tweet as rumor or not [2]. The querying tweets were searched and selected using a specially curated list of regular expressions. Example of one the regular expression used: `wh[a]*t[?!][?1]*`. The handcrafted list of regular expressions will lead to under generalization, and tweets with no response will be classified with a lot of uncertainty.

Another work on rumor detection was carried out by reference [3]. They used Conditional Random Fields (CRFs) that learn from the sequential nature of social media posts for rumor classification. They further created a dataset of tweets from Twitter on various topics with a mix of rumor and non-rumor tweets. This approach exploited both the tweet content and user data to classify a tweet yielding the best result so far. We use this as the baseline approach for our work. Various other works in this area have misdefined rumor, they consider rumor as a false piece of information without taking into consideration the time of posting, making them inconsistent with the established definition. Hence, we have no considered these works for our approach.

We propose a novel approach for detecting rumors on social media, the approach utilizes transformer models that are used as feature extractor, and whose output is feed into linear layers for classification of the tweets into rumor and non-rumor. The transformer models we used for this work are

1. BERT (Bidirectional Encoder Representation of Transformers) [4].
2. RoBERTa (A Robustly Optimized BERT Pretraining Approach) [5].
3. ALBERT (A Lite BERT) [6].
4. DistilBERT (A distilled version of BERT) [7].

The input to these models was a sequence of words, comprising source tweets and comments received on that source tweet. We utilized comments by various users to debunk a rumor as we believe that users exposed to rumors will try to find the veracity of the rumor using the comments. The collective and subjective sense-making via comments will lead to a conclusion regarding the veracity of the information. Using the approach and the models, we’re able to achieve state-of-the-art results.

This paper is organized into five sections. After the Sect. 1 of Introduction, Sect. 2 presents the Related work. Section 3 of this paper describes the Proposed approach and Sect. 4 shows the Results and the discussion about the results. Finally, the Conclusions are presented in the Sect. 5 of this paper.

2 Related Work

We have considered the works that have defined rumors as information that is unverified at the current moment in time. Due to this, few of the works done on this problem have not been considered.

Since it is extremely difficult to classify individual posts as a rumor or not, they try to deploy techniques to find the cluster of posts that are inquisitorial in nature and termed them as signal tweets [2, 8].

The approach taken in this paper uses three algorithms, each performing a particular task. The first algorithm identifies the newly emerging controversial topics based on the occurrence of signal tweets. The second algorithm gives regular expressions derived from the cluster of signal tweets and the third identifies features of signal clusters and this is used to rank the clusters by their probabilities of containing a rumor. A five-step procedure, undertaking the tasks of identification of signal tweets, identification of signal clusters, gathering detecting statements, capturing non-signal tweets, and finally ranking candidate rumor clusters are followed by the three aforementioned algorithms.

They successfully show that their novel approach of first filtering tweets with specific features, then clustering these tweets to detect statements in smaller clusters of tweets, and at last giving potential rumor statements has significant cost benefits over the traditional approach of first identifying trending topics and then identifying rumors. The primary areas of improvement include better filtering of enquiry and correction signals, automate updating of filtering patterns to reduce spamming, and adding features to create a large dataset of rumors and the associated logs for further analysis and development.

In the paper [3], the authors used CRFs as a classifier that allows for the aggression of tweets as individual posts [4]. They modeled the Twitter threads as a linear chain of graphs. The CRF takes graph $G = (V, E)$ as an input, it also takes into consideration the neighbors of each unit. The output to an input X will be a sequence of label Y , where y_i is not only based on features of x_i but also on the neighboring labels.

$$p(y|x) = \frac{1}{Z(x)} \prod_{a=1}^A \psi_a(y_a, x_a) \quad (1)$$

In Eq. (1), in the formula for the conditional distribution of CRF, $Z(x)$ is the normalization factor and Ψ_a is the set of factors in graph G . Hence, for the classification of a tweet, CRF exploits the sequence of rumor and non-rumor tweets leading

to the current tweet in question. Errors in earlier tweet classification in the sequence will then increase errors in subsequent tweet classification. The input features for this classifier were context features extracted from the tweet itself and the social features extracted from the user metadata provided in the dataset.

The paper [17] proposed a modified crow search algorithm (MCSA). This algorithm was an extended version of crow search algorithm and was used for extracting usability features from hierarchical model. The author in the research paper [18] used an improved boosting technique called DivBoosting for ensemble pruning.

3 Proposed Approach of Leveraging User Comments for Rumor Detection

For the purpose of this research, the PHEME dataset is selected [3]. The dataset contains tweets related to newsworthy events that can potentially lead to the propagation of rumors. The collected tweets are manually annotated by journalists. A tweet was labeled as a rumor when there was no evidence or no authoritative source had confirmed it at the time of posting. The dataset followed five different newsworthy events:

- Ferguson unrest: Protest due to shooting of an African American, Michael Brown, by a police officer on August 9, 2014.
- Ottawa Shooting: A Canadian soldier shot to death at Ottawa Parliament Hill on October 22, 2014.
- Sydney siege: Hostages were held at Lindt chocolate café at Martin Place in Sydney by a gunman on December 15, 2014.
- Charlie Hebdo: Gunmen entered the office of weekly newspaper Charlie Hebdo and killed and injured people working there on January 7, 2015.
- Germanwings plane crash: a passenger plane crash on French Alps on route from Barcelona to Dsseldorf on March 24, 2015.

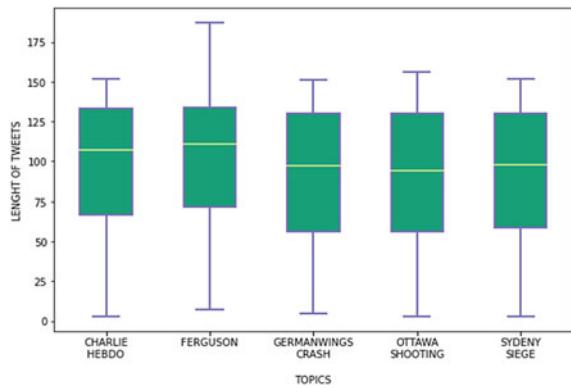
The dataset contains 5,802 manually labeled tweets of which 1,972 are labeled as rumors and 3,830 as non-rumor. These tweets are distributed differently across five newsworthy events as shown in Table 1. The dataset also contains the metadata of the user concerning each tweet. The tweet length, i.e., the number of words comprised in a tweet varies across the events. A boxplot of the word length concerning each event is shown in Fig. 1.

A. Hypothesis

User comments are an important aspect for classifying the veracity of a tweet. It was observed that the users tend to support unverified rumors in earlier stages, but there is a shift to debunking false rumors as time passes by [1, 16]. The collective effort of users on social media sites to determine the veracity of a tweet can be used with the source tweet content by a deep learning classifier to classify tweets.

Table 1 Tweet distribution in the pheme dataset

Event	Rumors	Non-rumors	Total
Charlie Hebdo	458(22.0%)	1,621(78.0%)	2,079(35.8%)
Ferguson	284(24.8%)	859(75.2%)	1,143(19.7%)
Germanwings crash	238(50.7%)	231(49.3%)	469(8.1%)
Ottawa shooting	470(52.8%)	420(47.2%)	890(15.3%)
Sydney Siege	522(42.8%)	699(57.2%)	1,221(21.1%)
Total	1,972(34.0%)	3,830(66.0%)	5,802

Fig.1 Tweet length boxplot concerning each event

The source tweet content cannot be used in isolation as some tweets can be very hard to classify while other tweets can be very easy. Take into consideration “The name of the police officer who fatally shot the kid would be reportedly announced by the police later on the day”, “reportedly” expresses uncertainty at the time of posting these tweets and hence, this tweet can be easily classified as a rumor. On the other hand tweets such as “10 people have died in a shooting at the Paris HQ of French weekly Charlie Hebdo, reports say” cannot be easily classified as rumor or non-rumor. So, we cannot completely rely on the source tweet itself, rather we will need to take into consideration the user comments.

B. Transformers

A transformer is an encoder–decoder model with self-attention to better understand long-range dependencies between input sequences [9]. In Fig. 2, the input is first fed to encoder, and the output from this is used by decoder. The encoder and decoder can be stacked on top of each other, and this is one of the hyper-parameter of the model. An input goes through all the encoder blocks, the final layer of the encoder block then transmits the representation to all the decoder blocks. The encoders and decoders are identical to each other, and they both contain multi-head attention module and on top a feed-forward network whose output is then transferred to the next encoder or

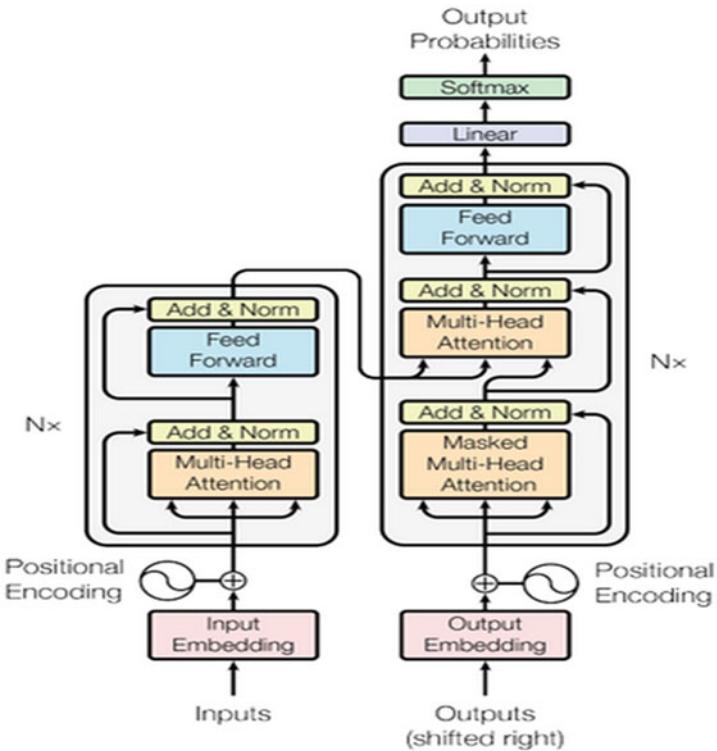


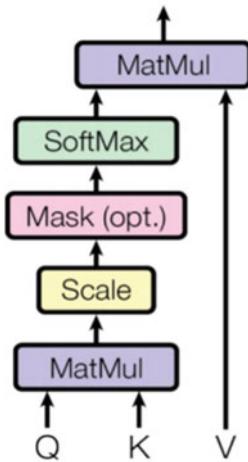
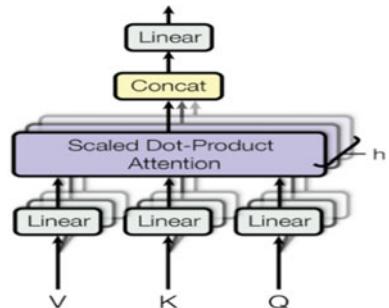
Fig. 2 Transformer architecture

decoder's multi-head attention in the sequence. They also contain skip-connections so information is not lost during training.

As this model has zero recurrent networks and as the entire sequence is feed to the model at once the position of each word will be lost. So, to solve this problem positional encoding is implemented right after the embedding. For each input sequence, the positional encoding is generated which gets added to the output of the word embedding.

A multi-head attention module (Fig. 4) is the combination of single-head attention modules (Fig. 3) used to reduce the miscalculation by single-head attentions. In single-head attentions, the module calculates attention scores related to each word (using Eq. (2)). This score gives details on the importance of other words in the sequence concerning a word. This score is obtained via training three vector parameters: Query, Key, and Value vectors.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

Fig. 3 Single-head attention**Fig.4** Multi-head attention

In multi-head attention, many such single-head attentions are used to calculate the attention scores. The output of each single-head attention is concatenated into a weight matrix W_0 , this is also a trainable vector. It is trained to get the final output of a certain size. This helps to capture all the attention that can possibly be attributed to a word concerning other words, reducing the miscalculation by a single attention head.

The attention module in the decoder stack is modified so the model cannot look ahead in time, this module is called masked multi-head attention. This modification will make sure that the output at position i depends on the input at a position less than i . The output from the decoder is feedback to the network in the next step. The output is also combined with the encoder output and fed to a multi-head attention module. This is useful to the decoder as it will keep the attention of the decoders on relevant parts in the input sequence. The output from this multi-head attention passes through the linear layer with softmax activation to yield output tokens/words.

C. Classifiers

For the rumor detection task, we will be using BERT, RoBERTa, ALBERT, and DistilBERT as feature extractors followed by linear layers as our classifiers.

BERT (Bidirectional Encoder Representations for Transformers) is a transformers model that uses attention mechanism to learn contextual meaning for a given text. It is Bidirectional in nature, i.e., it will read the entire sequence of words at once, unlike directional models that read input text sequentially in one direction. This helps the model to understand the context of each word in relation to the words around it, as it can look in both directions simultaneously. BERT uses the wordpiece tokenization technique, tokenization is the method to convert inputs into smaller units called tokens. In the wordpiece tokenization technique, the vocabulary initially contains all possible characters in the text and then the most frequent combinations of the existing characters and terms are combined iteratively [10]. BERT model was trained on two strategies:

1. Masked language model: This model masks 15% of the input tokens randomly and this training aims to make the model predict the masked tokens [4]. This allowed for the training of deep bidirectional transformers due to a better understanding of the context.
2. Next sentence prediction (NSP): The objective of this training was to determine whether the second sentence given as input was the actual next sentence from the training document or not. In the training dataset, 50% of the sentence pairs are actually the sentence pair in the training document.

RoBERTa (Robustly Optimized BERT Pretraining Approach) is a language model that replicates the BERT but greatly improves the performance. The performance increase can be attributed to the following reasons:

1. Longer training time, with larger batch size and dataset.
2. Next sentence prediction objective was removed.
3. Longer sequences were used for training.
4. Masking patterns applied were dynamically changed to the training data.
5. Using a byte-level tokenization technique like the Byte-Pair Encoding tokenization technique. In this tokenization technique, bytes are used to represent characters. This helps in tokenizing every single character in the input sequence without the need for an unknown token [11].

ALBERT (A Lite BERT) is a language model that is based on the BERT model but has 18 times fewer parameters and can train 1.7 times faster compared to BERT. This is achieved using the following two-parameter reduction techniques:

1. Factorized embedding parameterization, by decomposing the vocabulary into two small matrices.
2. Cross-layer parameter sharing is done to reduce the growth of parameters as the depth of the network increases.
3. Another important change was the introduction of sentence order prediction (SOP), a self-supervised learning loss, to reduce the problems due to NSP loss.

These changes lead to ALBERT being scaled to a much larger configuration and still having lesser parameters than BERT. This model uses the sentencepiece tokenization technique, this technique takes in an input stream, includes space as a character that it can use, then uses BPE or unigram to construct appropriate vocabulary [12].

DistilBERT is a distilled version of BERT, a distilled version is much lighter and easier to train making it computationally cheaper to train and use. This is done using knowledge distillation that trains lighter models using distillation loss so as to preserve the features of a larger and heavier model. The model was able to retain 95–97% of the performance using 40% fewer parameters and 60% faster when compared with BERT. This model uses the wordpiece tokenization technique for tokenizing the input sequence.

D. Transfer learning

It has been observed that humans can transfer knowledge acquired while doing one task (also known as source task) to solve another task (also known as target task). This can also be achieved in neural networks where a neural network is trained to do some task A, also known as source task, then it can also be used to perform a similar task B, known as target task.

A transfer learning is known as Inductive transfer learning if the following requirements are met [13]:

1. The source and target task should be independent of each other.
2. The dataset used by both the tasks must be independent.

We have applied inductive transfer learning on pre-trained BERT, RoBERTa, ALBERT, and DistilBERT, down-streaming them to perform the classification of tweets [14].

E. Approach

We propose to utilize user comments with the source tweets as the input sequence which is feed to the classifiers. Tweet preprocessing is carried out before it is feed to the classifier. In the preprocessing step, from every single tweet the hashtags, URLs, images, mentions, emoji, and abbreviations are removed. These features will not be utilized by the classifiers as they are unable to understand URL, images, mentions, emoji, and abbreviations. Consider the following tweet “BREAKING: 10 reportedly shot dead at Paris HQ of French weekly Charlie Hebdo <http://t.co/5F1D0wzoCQ>”, after tweet preprocessing the outcome will be “BREAKING: 10 reportedly shot dead at Paris HQ of French weekly Charlie Hebdo”.

After preprocessing the tweets, the source tweet is concatenated with the user comments tweets. This becomes the input sequence which is then fed to the tokenizer where only the first 512 tokens are selected. If the input sequence is less than 512 tokens then padding is applied. We apply the tokenizing technique based on the classifier model used. Once this is achieved, this input sequence is feed to the classifiers where it will first go through BERT, RoBERTa, ALBERT, or DistilBERT which will

Table 2 Hyper-parameter values for classifiers

Classifier	Learning rate	Batch size
BERT	0.00001	6
RoBERTa	0.0000002	4
ALBERT	0.00000002	4
DistilBERT	0.00002	8

extract features, and then these extracted features are feed to a linear layer for tweet classification.

The classifiers are trained using Adam optimizer [15] with varying learning rates and batch size. Table 2 shows the learning rate and batch size adopted by each classifier.

For the purpose of training, we have used the five-fold cross-validation method. In this method, we train the model on four events and evaluate the reminder event. This is done to simulate a real-life situation where the model can be trained on the previously observed set of tweets, and then this can be used to evaluate the new unseen tweets into rumor and non-rumor.

For the evaluation purpose of the proposed approach, we have used three different metrics:

1. Precision: It is the ratio of accurately predicting positive class to the overall positive class predicted by the model. High precision is related to the low false positive rate.
2. Recall: It is the ratio of accurately predicting positive observations to a number of observations in the positive class.
3. F1 score: It is a way to relate both precision and recall. This relation is nothing but the weight average. This a much better metrics than accuracy when there is uneven class distribution in the training dataset

4 Results and Discussion

The results of this method are given in Tables 3, 4 and 5, the evaluation is done

Table 3 Evaluation by event

Event	BERT			RoBERTa		
	P	R	F1	P	R	F1
Charlie Hebdo	0.76	0.81	0.78	0.74	0.67	0.69
Ferguson	0.67	0.57	0.57	0.66	0.54	0.51
Germanwings crash	0.71	0.71	0.71	0.68	0.68	0.68
Ottawa shooting	0.81	0.80	0.79	0.76	0.70	0.67
Sydney siege	0.75	0.71	0.71	0.76	0.76	0.76

Table 4 Evaluation by event

Event	DistilBERT			ALBERT		
	P	R	F1	P	R	F1
Charlie Hebdo	0.73	0.73	0.73	0.72	0.75	0.73
Ferguson	0.63	0.60	0.61	0.66	0.60	0.61
Germanwings crash	0.64	0.62	0.60	0.67	0.67	0.67
Ottawa shooting	0.80	0.78	0.77	0.76	0.74	0.73
Sydney siege	0.75	0.74	0.74	0.76	0.76	0.76

Table 5 Evaluation by event

Baseline model—CRF			
Event	P	R	F1
Charlie Hebdo	0.54	0.76	0.63
Ferguson	0.56	0.39	0.46
Germanwings crash	0.74	0.66	0.70
Ottawa shooting	0.84	0.58	0.69
Sydney siege	0.76	0.38	0.51

by events. Our approach was able to greatly outperform the CRF model (baseline) proposed by reference [3]. Our proposed approach outperforms when F1 scores are under consideration when compared to the baseline model. Moreover, our classifiers have better-balanced precision and recall when compared to the baseline. The baseline model sometimes outperforms our proposed classifier but, in those cases, our classifiers outperform in case of recall and F1 score. These results confirm that our approach, i.e., context learned by the classifiers via source tweet and user comment tweets can lead to a better rumor classification.

Rumor detection is the first component for a system that deals with rumors: (1) rumor detection, (2) rumor tracking, (3) rumor stance classification, and (4) rumor veracity classification [1]. Higher metrics in the first component can improve the overall performance of such systems. The main use of our approach can be to inform social media users whether the content they are currently viewing has been verified at the time of posting or not. This can greatly hinder the propagation of rumors as either the systems will flag such tweets and try to stop their propagation or the users will themselves try not to retweet such posts.

Our approach has greatly improved metrics when compared to the baseline due to the power provided by the language models that have been utilized for feature extractions. One of the reasons for higher metrics is because the language models utilized in our experiment can understand the context of a given input sequence much better when compared to the baseline model. The input sequence contains both the source tweet and the user comment tweets.

Our proposed approach may not perform as intended for early stage rumor detection as it might not be able to yield a good result when no user comment has been

observed. Standalone tweets are hard for even humans to determine as rumor or not, the performance of the model will greatly decrease if only source tweet is used for rumor classification. The proposed approach is also hindered by the fact that language models cannot take a very large input sequence due to computational expense because of the attention mechanism. The use of language models as feature extractors means that the model metrics will generally increase as the overall training dataset size increases. This means, with increases annotated data these language models will extract better features from the input and that will lead to better overall metrics.

5 Conclusion

In this research paper, we have introduced a novel approach for rumor detection by leveraging user comments and experimenting with four transformer-based classifiers on five news events collected from Twitter. Our approach has outperformed the baseline model under consideration for rumor detection and we have proven this by training various models. Our approach achieves superior performance in terms of the F1 score and is optimized to show better balance for precision and recall. Our approach greatly increases the accuracy of the rumor detection step in the rumor classification system.

User-Generated Content plays a central role in making various decisions, like political policies, journalism, etc. Also, during natural disasters, it is user-generated content that can help spread vital information and it can also be used to prevent chaos during these events. Currently, we do not have a classification system that can reliably label various content as rumor or non-rumor, and veracity of the content. If users consume unverified content, they can be solicited into a larger conspiracy that can greatly damage human society. Hence, it a necessity to develop tools that can aid the rumor classification system. This work shows an increase in accuracy by using a novel approach which will lead to an increase in the overall accuracy of the classification systems and make the internet a more reliable and dependable source of information.

References

1. Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, Rob Procter, Detection and resolution of rumours in social media: a survey (April 2017)
2. Zhe Zhao, Paul Resnick, Qiaozhu MeiQiaozhu Mei, Enquiring minds: early detection of rumors in social media from enquiry posts (May 2015)
3. Arkaitz Zubiaga, Maria Liakata, and Rob Procter, Exploiting context for rumour detection in social media (Sep 2017)
4. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding (May 2016)

5. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach (July 2019)
6. Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut, ALBERT: a lite BERT for self-supervised learning of language representations (Sep 2019)
7. Victor Sanh, Lysandre Debut, Julien Chaumond, Thomas Wolf, DistilBERT, a distilled version of BERT: smaller, faster, cheaper, and lighter (Oct 2019)
8. M. Mendoza, B. Poblete, C. Castillo, (Twitter under crisis: Can we trust what we rt? (July 2010)
9. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin, Attention is all you need (Jun 2017)
10. Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey et al., Google's neural machine translation system: bridging the gap between human and machine translation (Sep 2016)
11. Rico Sennrich, Barry Haddow, Alexandra Birch, Neural machine translation of rare words with Subword units (Aug 2015)
12. Taku Kudo, John Richardson, SentencePiece: a simple and language independent subword tokenizer and detokenizer for neural text processing (Nov 2018)
13. Liu Yang, Steve Hanneke, Jaime Carbonell, A theory of transfer learning with applications to active learning
14. Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierrick Cistac, et al., Transformers: state-of-the-art natural language processing (Oct 2020)
15. Diederik P. Kingma, Jimmy Ba, Adam: a method for stochastic optimization (Dec 2014)
16. Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, Peter Tolmie, Analysing how people orient to and spread rumours in social media by looking at conversational threads (Nov 2015)
17. Deepak Gupta, Joel J. P. C. Rodrigues, Shirsh Sundaram, Ashish Khanna, Valery Korotaev, Victor Hugo C. Albuquerque, Usability feature extraction using modified crow search algorithm: a novel approach. *Neural Computing and Applications* (Springer), (2018), <https://doi.org/10.1007/s00521-018-3688-6>
18. Mohammad Mehdi Motahari Kia, Jafar A. Alzubi, Mehdi Gheisari, Xiaobo Zhang, Mohamad Taghi Rahimi, and Yongrui Qin, A novel method for recognition of persian alphabet by using fuzzy neural network. *IEEE Access*, vol. 6 (2018)
19. Jafar A. Alzubi, Diversity-based boosting algorithm. *Int. J. Adv. Comp. Sci. Appl.* **7**(5) (2016)
20. Omar A. Alzubi, Thomas M. Chen, Jafar A. Alzubi, Hasan Rashaideh, Nijad Al-Najdawi, Secure channel coding schemes based on algebraic-geometric codes over hermitian curves. *J. Univer. Comp. Sci.* **22**(4) (2016)

A Cost-Efficient QCA XOR Function Based Arithmetic Logic Unit for Nanotechnology Applications



Divya Tripathi and Subodh Wairyा

Abstract Quantum-dot Cellular Automata (QCA) is an innovative nanometer technology that suggests less dimension, less power consumption, with extra speed, and reflected as an amplification to the scaling complications with CMOS computation methodology. Additionally, an Exclusive-OR (XOR) gate is a basic building block in various significant circuits like a Full Adder (FA), comparator, etc. In this work, we designed an optimized QCA XOR gate and its application in some logical gate designing. This paper shows the cell optimization and a realization of the QCA XOR gate, full adder, and full subtractor and further these layouts are used to design 1-bit QCA ALU. Full adders are the most critical component of any electronic device because all electronic devices need ALU and full adders are the primary architecture of an ALU. The optimized full adder has been chosen to build QCA ALU and parameters like number QCA cells, latency, area, and quantum cost have improved in the proposed design approach and the proposed adder represents the least area and latency among other reference circuits in nanotechnology. All the simulations are verified with the free accessible QCA Designer simulation environment.

Keywords Nanotechnology · Quantum-dot cellular automata · ALU · QCA Designer

1 Introduction

Quantum-dot Cellular Automata (QCA) are a non-transistor computation methodology that encrypts binary information through an arrangement of charges between quantum dots. The imperative QCA logic prerequisites are the majority and inverter gates which can be used to plan several QCA circuits. Quantum-dot Cellular Automata (QCA) technology is appreciated due to its less power consumption, fast speed, and a minor dimension, and so, it is an encouraging substitute to CMOS technology.

D. Tripathi (✉) · S. Wairyा

Electronics & Communication Engineering, Institute of Engineering & Technology, Dr. APJ. Abdul, Kalam Technical University, Lucknow, India

Conventional CMOS has conquered the manufacturing industry for the past few decades, but now we live in a highly erudite era regarding technologies so our requirements for higher speed, reduced area, and low power cannot be fulfilled by customary CMOS technology. The issue of Power Dissipation and Leakage Current cannot be astounded by additional scaling of CMOS parameters. It is better to shift near-new alternatives to IC Designing. Quantum-dot Cellular Automata (QCA) based on interacting Quantum dots, which are more appropriate for logic circuits with higher and lower power scattering at the nanometer scale is possibly the best alternative among all. In the near future, QCA can take over CMOS because of its significant benefits. In this paper primarily design XOR gate and implementation of a proposed XOR gate in its application of various logic gates such as Full Adder, Full Subtractor, and 1-bit ALU by using the Quantum-dot Cellular Automata (QCA) designer tool and these proposed designs are improved than the previously reported designs regarding the area as well as latency. QCA is the prospective addition of CMOS technology because the simple influence of quantum phenomena on the ever tiny transistor operation will not let more reductions of parameters in CMOS technology. QCA originated first in 1993 by lent et al., and was substantially proved in 1997 [1–4]. It can produce extraordinary device density, less power consumption, and speedy switching. QCA suggested a new computing pattern in nanotechnology.

QCA structures are assembled as an arrangement of quantum cells within which each cell has communicated electrostatically with its adjacent cells. QCA exploits an innovative form of computation, where the state of an electron rather than the customary current contains the binary information. As an alternative to interrelating wires, the cells transfer through the circuit. This article has the basic contribution are as follows:

- (a) Designing an ultra-efficient 2-input and 3-input XOR gate (N7 and N11) using QCA.
- (b) Designing of an efficient full adder using a proposed cost-efficient XOR gate in QCA.
- (c) Designing of an optimized full subtractor using a proposed cost-efficient XOR gate.
- (d) Designing of an efficient 1-bit QCA ALU using proposed logical and arithmetic QCA layouts.
- (e) The proposed architectures are compared to the existing designs based on quantum cell count, area, latency, and the quantum cost which confirms that proposed designs have a lesser area and faster speed as related to their previous best counterpart.

The remainder of the paper is as follows; Section 2 defines the outline of QCA technology. Implementation of QCA design and Simulation outcomes of XOR topology, Full Adder, and Full Subtractor and 1-bit ALU are discussed in Sect. 3. Section 4 explained about simulation result and discussion of the proposed design. The conclusion is in Sect. 5.

2 A Brief Oversight of Quantum-Dot Cellular Automata

QCA formations are putting together an arrangement of quantum cells within which each cell has communicated electrostatically with its neighboring cells. QCA technology has built-in capacities for digital circuitry schemes without ensuring any Boolean function [3–6]. QCA cell plays a vital role in QCA methodology, which allows us to implement a couple of computations and transfer the data through the overlap. A simple QCA cell contains a speculative squarish space that holds the four energy sites there in which electrons can absorb. Each cell is invaded by two electrons. Also, the Columbic interaction between electrons can create two distinct cell states with different charge arrangements [5–8]. This arena for an electron is meant by a dot in the cubicle cell. Quantum mechanical tunneling barriers are used to couple the phases so that electrons can tunnel via them determined by the structural state. Columbia's repulsion compels the electrons to involve the furthermost dots in a QCA cell which resembles the nethermost energy state owned of the circuitry. Cell polarization refers to the relative locations of the electrons in a cell and it selects whether it is signifying binary "1" or "0". Two categories of QCA cells are there the first is 90° and the next is 45° cells. Figure 1 presents a 90° cell with the state of polarization of $P = +1$, which signifies binary 1 [9]. The clock signal delivers the desirable force to do the calculation. A simple QCA cell contains a speculative squarish-shaped arena in which there are four energy sites in which electrons can absorb. This arena for an electron is signified through a dot in the cubicle cell. The stages are coupled over quantum mechanical tunneling barriers and electrons can tunnel via them dependent on the estate of the structure. Coulmbia's repulsion compels the electrons to reside

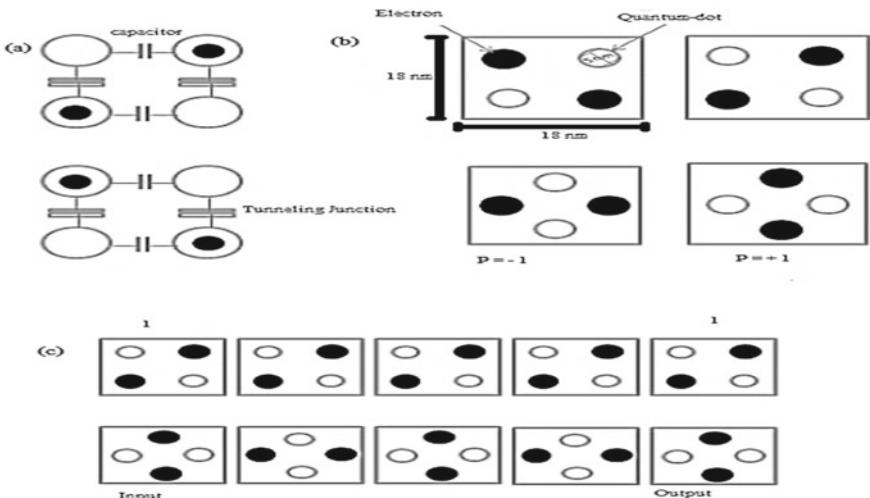
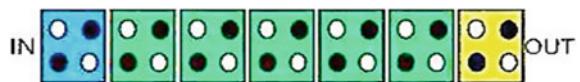


Fig. 1 QCA Cells **a** Functional outline **b** Polarizations in cells **c** QCA wire

Fig. 2 Transmitting binary “1” in QCA wire



in the farthermost dots in a cell which resembles the nethermost energy estate of the structure.

QCA cells are organized in a continual fashion to compose a QCA Wire. The attractiveness and repulsiveness taking place because of Columbian force between the end-to-end cellular sources the cell polarization so that it can arrange in a communication channel according to its neighboring cells so the communication of data besides the cell array takes place. QCA wires communicating logic “1” using 90° QCA cells are presented in Fig. 2. In Fig. 2, the input cells are enforced through an outside source and are powerfully polarized in one way. The input cells drive another QCA cell in the NULL state which tends to bring into line them to the polarization of the input cell to attain the structure’s ground stated [10].

2.1 Majority Gate

The arrangements of the majority gate are offered in Fig. 3. The output Y is well defined as $Y = PN + PO + NO$. Output QCA cell of the gate diverges affording to the calculations of a QCA cell in the medium of the gate. Straight off this output “Y” as it may be driven out along the help of QCA wire whichever can execute as an input to other gates. The majority gate plays a significant role to construct the OR/AND gates [5, 6]. If one input is stable to 1/0, the subsequent function Y is the OR/AND of the rest of the two inputs.

$$M(P, N, O) = F = (P * N) + (N * O) + (P * O) \quad (1)$$

where P, N, and O are 3 inputs of the majority gate and F is the one output of the majority gate. Basic digital logic gates are made because of the majority gates as

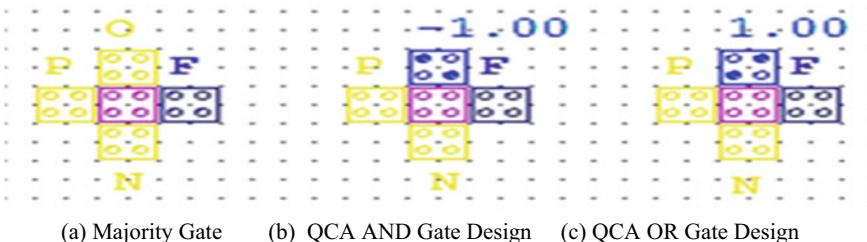


Fig. 3 Basic QCA Logic gates Architecture. **a** Majority gate. **b** QCA and gate design. **c** QCA OR gate design

shown in Fig. 3. The basic architecture of the majority gate contains 5 quantum cells in which 3 cells used as input 1 cell are used as output and 1 cell is fixed normal cell, we can change the polarization of any input cell if we will assign polarization of one input cell is -1 so whole architecture works as AND gate and if the polarization of the cell is +1 then whole architecture become OR gate as presented in Fig. 3.

2.2 QCA Clocking

The data streaming in the QCA architecture is managed and coordinated via the clocking mechanism. Clocking offers the ability to make and keeps off the meta-stable phase [11]. Concerning QCA cell, the meta-stable phase-run parallels the polarization, cells that may not be characteristically predictable as logic 1 or logic 0. Data is organized by the clocking signal and input and signified via the polarization of cells [12].

The clocking in QCA technology is not similar as they are customary CMOS circuits, the QCA clocking arrangements contain quadruple phases: Switch (unpolarized cells determined by several inputs and become polarized dependent on their neighbors' polarization), control (cells contained in the same binary state so that it can apply as per the input to other cells), release (barriers are taken down and cells remain unpolarized), and relax (cell remains unpolarized) [13, 14]. This is the phase divergence of the quarter cycle of all these clocking phases as it may be supported through generating 4 clocks each one with $\pi/2$ phase difference from the previous one shown in Fig. 4.

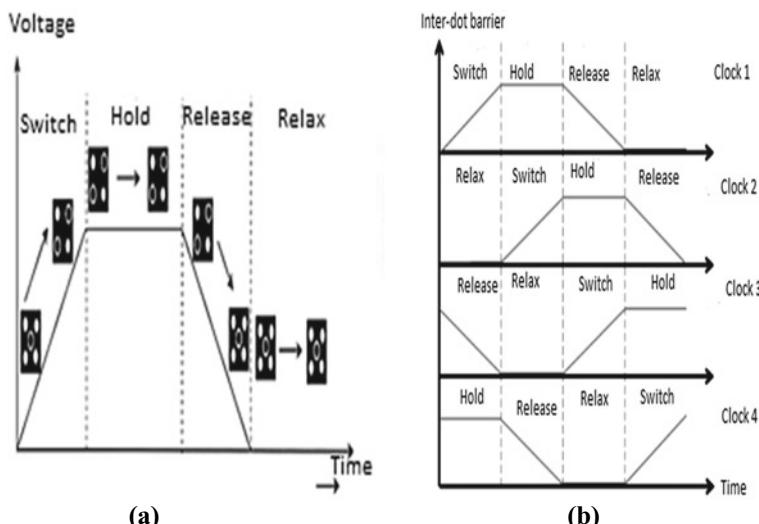


Fig. 4 Clocking concept in QCA **a** Clock zone of a QCA cell **b** Clocking in QCA: Four phase

2.3 QCA Designer

QCA Designer tool used for simulation of the complex QCA architectures. Predominantly it was produced in the ATIPS Laboratory, University of Calgary. The newest version of QCA Designer has three diverse simulation engines included. A piece of the three engines has a dissimilar and significant band of profits and disadvantages to boot, each simulation engine can do a systems comprehensive confirmation or a set of user-selected vectors. This tool is adapted for the estimation of energy dissipation in QCA-based layout also. The design of workability is tested under the simulation engine of the bi-stable approximation. The multilayer QCA layout is not feasible to design the circuitry as well as the practicability existence of these techniques is difficult to implement [13]. ATIPS Laboratory, Calgary University developed QCA Designer initially. It is used for the simulation of digital circuits by creating QCA circuit layouts. Three simulation engine is used in the latest version of QCA designer tool which is followed below.

1. Coherence Vector,
2. Coherence Vector (w/Energy), and
3. Bistable Approximation.

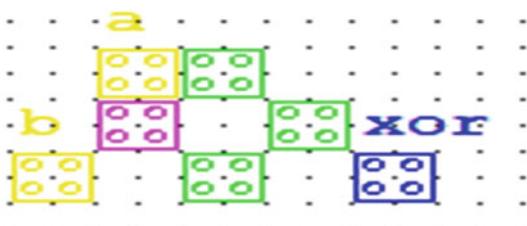
All three simulation engines above have their individual merits and demerits, but each of them can perform a comprehensive authentication of the layout or set of vectors designated by the end-user [14].

3 QCA-based Proposed Logic Design

3.1 Reviews on XOR Gate

In this segment, a digital circuit called the “XOR” gate is designed. This is a primary digital circuitry that is practiced in many types of combinatorial circuitry such as arithmetic logic circuits, multiplexers, comparators, and error detectors. Many researchers have presented effective and robust XOR designs in QCA nanotechnology. Numerous prior architectures of two inputs and three inputs QCA “XOR” gate design are presented in Fig. 5. Some prior layout designs of 2-input QCA “XOR”

Fig. 5 QCA layout of the proposed 2-input XOR Gate (N7)



gate are used for the literature survey. Niemier, M.T. [8] aimed that QCA XOR gate contains 60 number of cells, $0.011 \mu\text{m}^2$ areas, and 1.5 clocking zone latency. This architecture delivers a significant number of cells and delivers a great field. To solve these difficulties, Hashemi, S. et al. [9] planned a novel 2-input QCA XOR gate publicized with only 51 cells, $0.092 \mu\text{m}^2$ area, and 2 clocking latency. To diminution the cell count, one more configuration is proposed by Chabi, A. M. et al.[10] that contains 29 cells, $0.041 \mu\text{m}^2$ area, and 0.25 clocking latency. One more design offered to cut the expense of the QCA XOR gate by G. Singh, et al.[11], using two QCA inverters and five inputs QCA majority gate with 28 cells, an area of $0.035 \mu\text{m}^2$, and 0.75 clocking zones latency. The embodiment according to Bahar, A. N. et al.[12] has compact the cell count till 12 cells for 2-input and 3-input XOR gate, area $0.021 \mu\text{m}^2$, and latency of 0.05. According to Roohi, A. et al. [13] introduced XOR gate has 14 cells, $0.034 \mu\text{m}^2$, and 0.5 clock latency.

3.2 Proposed QCA Architecture of XOR Gate Topology

This segment describes a digital circuitry called Exclusive-OR (XOR) gate. It is a main digital circuitry that is practiced in many diverse kinds of computational circuits such as Arithmetic logic circuits, Multiplexer, Full adder, Comparators, and Error detector circuits. In summation, thus we can also use the Exclusive-OR (XOR) gate which is valuable to design any complex circuitry. In the literature, numerous QCA-designer-made XOR gates are shown in [8–15]; some of these designs are single layers and a few are multilayer designs.

In this section, an innovative design of a 2-input and 3-input QCA XOR gate has been introduced. The conventional and low complexity XOR gate had been introduced by so many researchers. But our proposed design is optimum as compared to its best previous counterparts. To investigate the effectiveness of our projected QCA XOR gate, numerous complex QCA design has been proposed later. The QCA layout of the 2-input and the 3-input QCA XOR gate is presented in Figs. 5 and 7, and simulation waveform is presented in Figs. 6 and 8, respectively (Table 1).

In this paper, an efficient and optimum design of 2-input and 3-input XOR gate using the QCA Designer tool is suggested. The proposed layout has very less cell count and high density as compared to its best existing counterpart. The new 2-input XOR gate design contains 7 numbers of quantum cells, an area of $0.012 \mu\text{m}^2$ and 0.25 latency and a 3-input XOR gate contains 11 number of quantum cells, and an area of $0.020 \mu\text{m}^2$ and 0.25 latency.

3.3 Proposed Full Adder Using Proposed XOR (N7) Gate

A full adder has 3 inputs, “A”, “B”, and “Cin” and two outputs “Sum” and “Carry”, as shown in Fig. 9 and the QCA layout of the proposed full adder is presented in

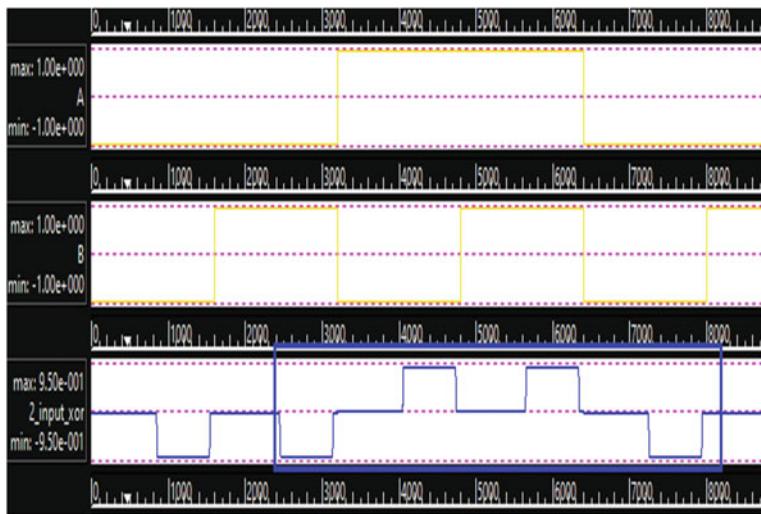


Fig. 6 Simulation waveform of the proposed 2-input XOR gate (N7)

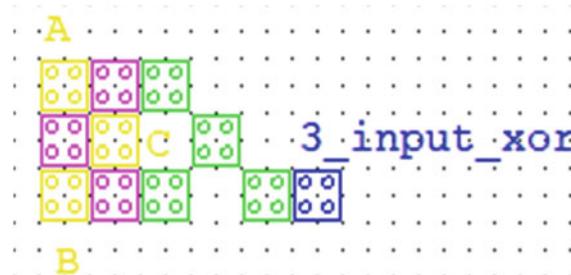


Fig. 7 QCA layout of the proposed 3-input XOR Gate (N11)

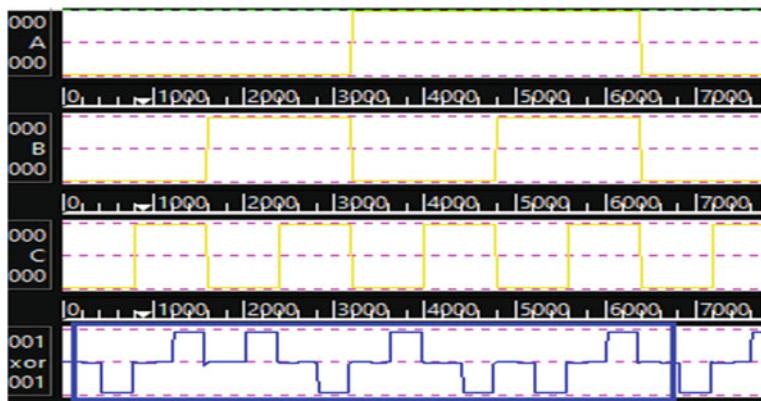


Fig. 8 Simulation waveform of proposed 3-input XOR gate (N11)

Table 1 Comparison table of various QCA XOR design

QCA XOR gate	No. of cells	Area (μm^2)	Latency (Clocking cycles)
[8]	60	0.011	1.50
[9]	51	0.092	2.00
[10]	29	0.041	0.25
[11]	28	0.035	0.75
[12]	12	0.021	0.05
[13]	14	0.034	0.50
Proposed 2-input XOR gate	7	0.012	0.25
Proposed 3-input XOR gate	11	0.020	0.25

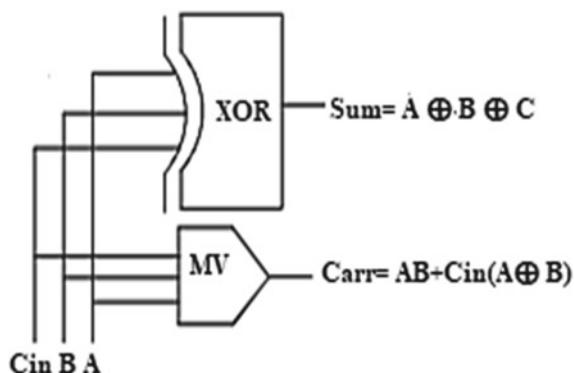
Fig. 9 Block diagram of the proposed QCA full adder

Fig. 10. The proposed QCA full adder contains 16 number of QCA cells which are lesser as compared to existing designs [16–18]. The generalized Boolean equations for “Sum” and “Carry” are given in equations below (Table 2):

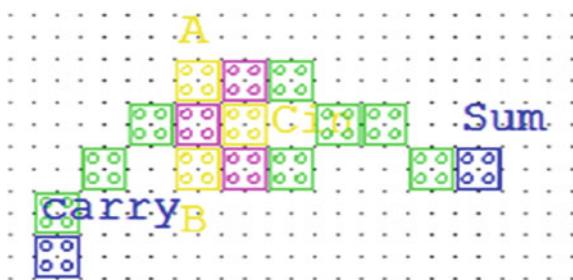
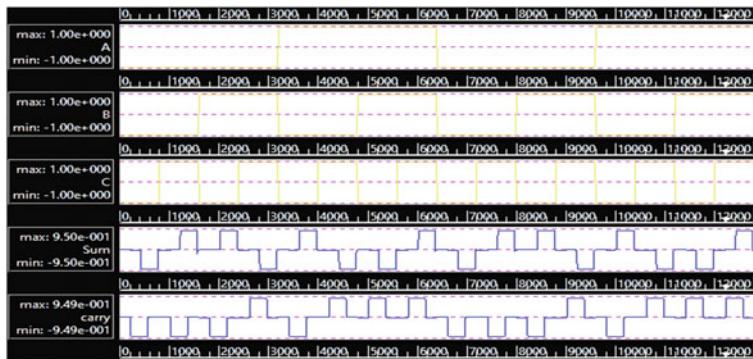
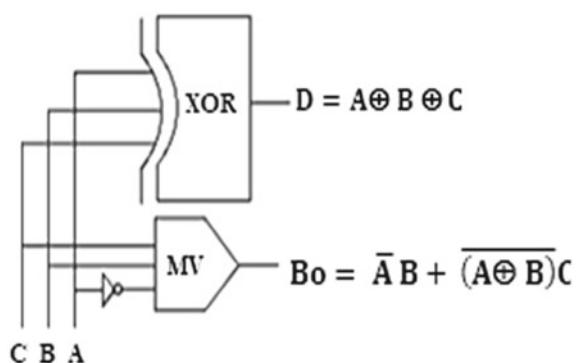
Fig. 10 QCA layout of the proposed Full Adder logic design (N16)

Table 2 QCA full adder design comparison

QCA full adder	No. of cells	Area (μm^2)	Latency (Clocking Cycles)
Previous design [16]	46	0.048	4.00
Proposed design	16	0.030	0.25

$$\text{Sum} = A \oplus B \oplus C$$

$$\text{Carr} = AB + \text{Cin} (A \oplus B) \quad (2)$$

**Fig. 11** Simulation waveform of the proposed QCA Full Adder gate (N16)**Fig. 12** Schematic diagram of full subtractor

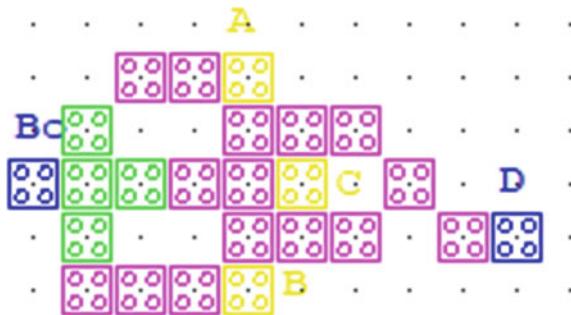


Fig. 13 QCA layout of the proposed full subtractor

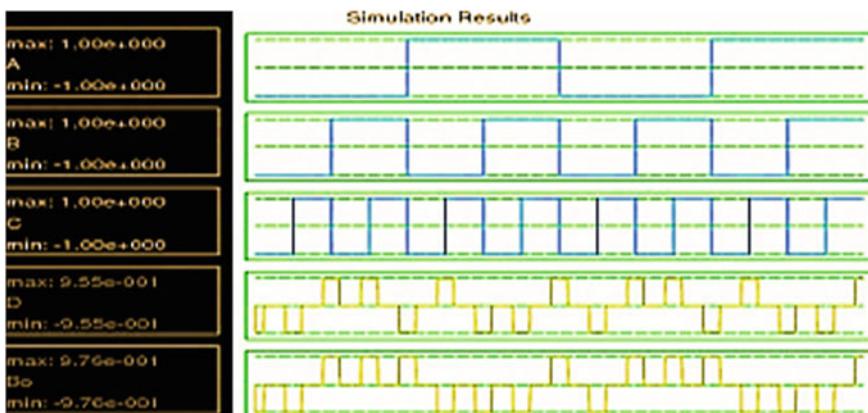


Fig. 14 Simulation waveform of the proposed QCA full subtractor

3.4 Proposed Full Subtractor Using Proposed XOR Gate (N7)

A full subtractor is a combinatorial circuitry that subtracts three bits and produces their difference. It is a combination of XOR gate, NOT gate, and AND gate, which gives the output as difference Diff and if a borrow is taken then it is shown by Borrow. The proposed QCA full subtractor contains only 24 numbers of QCA cells and $0.03 \mu\text{m}^2$ area, which is less as compared to previous designs [19]. The logic function for the full subtractor is (Table 3),

$$D = A \oplus B \oplus C$$

$$Bo = \overline{AB} + \overline{(A \oplus B)} C \quad (3)$$

Table 3 QCA full subtractor comparison

QCA full subtractor	No. of cells	Area (μm^2)	Latency (Clocking Cycles)
Previous design[19]	39	0.042	0.5
Proposed design	24	0.034	0.5

Table 4 ALU truth table

V2	V1	V0	Operations	Functions
0	0	0	$U = X - 1$	DECREMENT
0	1	1	$U = X + 1$	INCREMENT
0	0	1	$U = X + Y$	ADDITION
0	1	0	$U = X + Y' + 1$	SUBTRACTION
1	0	0	$U = X \wedge Y$	AND
1	0	1	$U = X \odot Y$	XOR
1	1	0	$U = X \vee Y$	XNOR
1	1	1	$U = X \vee Y$	OR

3.5 1-bit ALU Architecture

Arithmetic Logic Unit (ALU) is a vital block in many devices such as microprocessors, microcontrollers, and digital signal processors used in embedded system design in nanotechnology applications. ALU is the heart of the CPU as all the processing is done in the ALU [21]. Full adder is the most important structure of an ALU. Hence the overall performance of ALU majorly depends upon the full adder design. Energy-efficient full adder design is desirable for better ALU architecture and performance (Table 4).

The existing ALU [22–23] contains very few operations and uses multilayer architecture. The proposed architecture consists of two 4:1 MUX, 1-bit Full Adder, logic block, and 2:1 MUX. Signals V2V1V0 act as a control signal for performing various operations while X, Y, and Zin are the logic inputs to the ALU, SUM, U, and COUT are the output signals and coplanar architecture also. The truth Table 5 shows the performance of the ALU. Table 6 presents the logic operations with a number of cells and area comparison of QCA 1-bit ALU architecture and the proposed ALU logic block diagram as shown in Figs. 15 and 16 shown the QCA layout of 1-bit ALU.

Table 5 Logic operations with number of cell and area comparison

Logic operation	No. of cells	Area (μm^2)
XOR	7	0.012
XNOR	9	0.015
AND	5	0.003
OR	5	0.003
ADDITION	16	0.030
SUBTRACTION	19	0.500

Table 6 Proposed QCA digital logic designs

Sr. no	Proposed QCA logic design	No. of cells	Area (μm^2)	Latency (Clocking Cycle)	Quantum cost (Area * Latency)
1	2-input XOR	7	0.12	0.25	0.045
2	3-input XOR	11	0.02	0.25	0.005
3	Full Adder	16	0.30	0.25	0.075
4	Full Subtractor	24	0.35	0.50	0.175
5	1-bit ALU	391	0.54	2.00	1.081

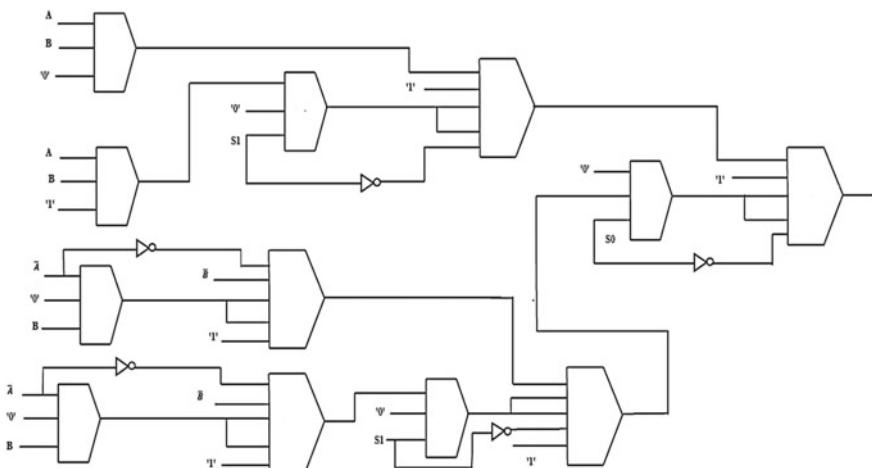


Fig. 15 Logic implementation diagram of the proposed 1-bit ALU

4 Simulation Result and Discussion

QCA Designer is taken for simulation of the QCA layout of the proposed designs. The simulation parameters like number of cells, area and latency (clocking cycles), and quantum cost have been measured and matched. In this paper, cell optimization is used to design efficient proposed digital logic design. Table 5 shows the all proposed

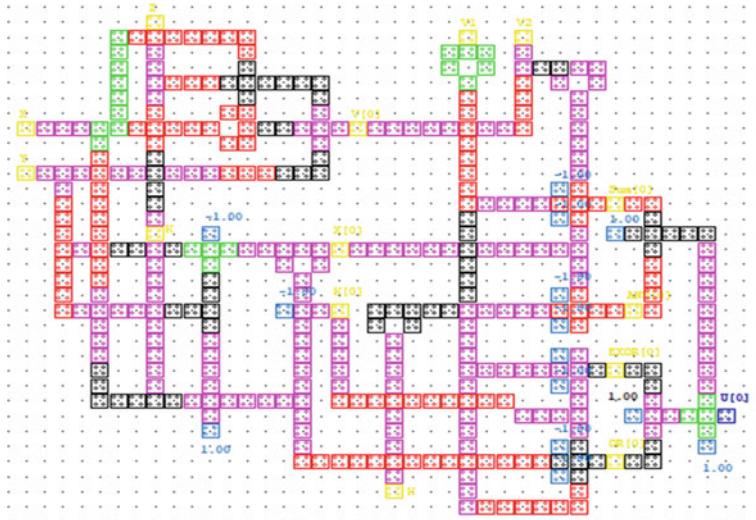


Fig. 16 QCA Architecture of the proposed 1-bit ALU

logic design with the parameters measured. It shows that our proposed designs gained less QCA cells, more capable in the parameters of the area, latency at the time than compared to previous existing designs.

5 Conclusion

Nanotechnology has been the center of current research that expands the horizon of nano-electronic devices in the past 10 years. As digital logic gates are rudimentary for maximum digital circuits, requiring high speed, less complexity, and minimum area designs are important. QCA is a worthy and reliable alternate to CMOS technology owing to less power consumption, more speed, and higher compactness. Conversely, a full adder is the crucial architecture in logical and arithmetical applications such as ALU. In this paper, we have proposed QCA-adder-based high speed and less power ALU architecture based on an optimal two-input and three-input Exclusive-OR (XOR) gate in quantum-dot cellular automata with minimum area, latency, and complexity as compared to its best existing counterpart. The proposed ultra-efficient XOR gate design utilizes just 7 normal QCA cells and has a latency of only 0.25 clock cycles. The proposed 2-input and 3-input XOR gate is applied for further carrying out of the optimum full adder, and full subtractor design and 1-bit ALU has 7 cells, 11 cells, 16 cells, 24 cells, and 391 cells, respectively. The proposed designs have abundantly decreased the area coverage, which demonstrates to be of an extraordinary performance when considering large and complex designs. In this work, we optimized and realized the QCA cell counts, reduced the area, latency, and

quantum cost of various proposed logic designs as compared to their best existing counterparts. In conclusion, the proposed methodology is an efficient way to reduce the parameters like QCA cells, area, and quantum cost also. The proposed designs can be used in most of the low-power and more speed VLSI applications.

References

1. C. Lent, P. Tougaw, A device architecture for computing with quantum dots. Proc. IEEE **85**(4), 541–557 (1997)
2. M.A. Shafi, A.N. Bahar, S.M. Shamim, K. Ahmed, Average output polarization dataset for signifying the temperature influence for QCA designed reversible logic circuits. Elsevier Inc. **340**, 42–48 (2018)
3. D. Tripathi, S. Wairy, An efficient energy ripple carry adder for nanotechnology applications. 3rd International Conference on VLSI, Communication & Signal Processing (3rd VCAS-2020), MNMIT, Allahabad, October (2020)
4. D. Tripathi, S. Wairy, An energy efficient magnitude comparator for nanotechnology applications. IJRTE **8**(6), 430–436 (2020)
5. K. Sridharan, V. Pudi, Design of arithmetic circuits in quantum dot cellular automata nanotechnology. Book Stud. Comput. Intell. **599** (2015)
6. A. Moustafa, Efficient quantum-dot cellular automata for half adder using building block. Quant. Inf. Rev. **7**, 1–6 (2019)
7. M.T. Niemier, Designing digital systems in quantum cellular automata. M.S. thesis, University of Notre Dame (2004)
8. S. Hashemi, R. Farazkish, K. Navi, New quantum dot cellular automata cell arrangements. J. Comp. Theo. Nanosci. **10**, 798–809 (2013)
9. A. Chabi, S. Sayedsalehi, S. Angizi, Efficient QCA exclusive-or and multiplexer circuits based on a nanoelectronic-compatible designing approach. Hindawi Publishing Corporation International Scholarly, vol. 9, (2014)
10. G. Singh, R.K. Sarin, B. Raj, A novel robust exclusive-or function implementation in QCA nanotechnology with energy dissipation analysis. J. Comput. Elect. **15**, 455–465 (2016)
11. A.N. Bahar, S. Waheed, N. Hossain, Md. Asaduzzaman, A Novel 3-input XOR function implementation in quantum dot-cellular automata with energy dissipation analysis. Alexand. Eng. J. **57**, 729–738 (2017)
12. A. Chabi, A. Roohi, H. Khademolhosseini, S.S. Sheikhfaal, Angizi, R.F. DeMara, Towards ultra- efficient QCA reversible circuits. Microproc. Microsyst. **49**, 127–138 (2017)
13. D. Fengbin, X. Guangjun, Z. Yongqiang, P. Fei, A novel design and analysis of comparator with XNOR gate for QCA. Microp. Microsys. **55**, 131–135 (2017)
14. Wang, Lei, Xie, Guanjun, A novel XOR/XNOR structure for modular design of QCA Circuits. Trans. Circuits Syst. **II**, 1–5 (2020)
15. S. Zoka, M. Gholami, A novel efficient full adder–subtractor in QCA nanotechnology. Int. Nano Lett. **9**, 51–54 (2019)
16. D. Mokhtari, A. Rezai, H. Rashidi, F. Rabiei, S. Emadi, design of novel efficient full adder circuit for quantum-dot cellular automata technology. Elect. Energ. **31**, 279–285 (2018)
17. R. Chakrabarty, N.K. Mandal, Design of a controllable adder-subtractor circuit using quantum dot cellular automata. IOSR J. Elect. Elect. Eng. **2**, 44–59 (2017)
18. A.N. Bahar, S. Waheed, N. Hossain, Md. Asaduzzaman, A novel 3-input XOR function implementation in QCA with energy dissipation analysis. Alex. Eng. J. **57**(2), 729–738 (2018)
19. M.H. Moaiyeri, F. Sabetzadeh, S. Angizi, An efficient majority based compressor for approximate computing in the nano era. Microsyst. Technol. **24**(3) (2018)
20. B. Debnath, J.C. Das, D. De, Design of 3:2 compressor using quantum dot cellular automata. IEEE VLSI Device Circuit and System (VLSI DCS) (2020), pp. 304–308

21. Roy, R., Sarkar, S., Das, S.: Multilayer structure of a new portable, low-cost and reversible arithmetic and logic unit using high-speed and low-power qca technology with good-scalability. *International Journal of Recent Tchnology and Engineering.* 10568–10575 (2019).
22. S. Ahmadpour, M. Mosleh, S. Rasouli Heikalabad, The design and implementation of a robust single-layer QCA ALU using a novel fault-tolerant three-input majority gate. *J. Supercomput.* **76**, 10155–10185 (2020)

Forecasting of PM10 Using Intelligent Crow Search Algorithm Tuned Feed-Forward Neural Network



Shalini Shekhawat, Akash Saxena, A. K. Dwivedi, and Vishal Saxena

Abstract Pollution forecast is a pioneering task and considered as a preliminary action taken by city planners as it can exactly locate the location of industrial plants and other development centers. Along with that on the basis of pollution profile, major decisions can be taken for controlling and combating it. Keeping this fact in mind, we propose a supervised structure of Feed-Forward Neural Network for predicting PM10 concentration in Jaipur city. For training the net, we employ our recently proposed algorithm based on the intelligent behavior of crow named as Intelligent Crow Search Algorithm (ICSA). It is observed that the developed ICSA yields better results when tested on the unknown samples. Comparative analysis of ICSA-based networks has been carried out with other contemporary nature-inspired algorithm tuned networks.

Keywords Crow search algorithm · Air Pollution · PM 10

1 Introduction

Air pollution is one of the most challenging threats appearing in front of all the countries. Deforestation, industrialization, increasing number of vehicles, and continuously spreading urban areas are increasing this problem exponentially. In India, almost all the metropolitan cities are facing the severe results of air pollution which are now moving toward smaller cities also. Air pollution is a result of the combination of toxic gases emitted from different industries and vehicle emissions. There are many ambient respirable particles present in the air which can affect the human

S. Shekhawat (✉) · A. Saxena
SKITM&G, Jaipur, India

A. K. Dwivedi
Rajasthan Technical University, Kota, India

V. Saxena
JECRC, Jaipur, India
e-mail: vishalsaxena.math@jecrc.ac.in

respiratory system severely. A long-term contact with these particles is even more hazardous as it results in cardiovascular disease and bronchitis [1].

These suspended particles (PM10 and PM2.5) [2] are easily inhalable due to their size, which is smaller than 10 μm and attracted a lot of focus of researchers as their adverse effects on human health. PM10 is formed by a wide variety of chemical substances including organic and inorganic compounds, metals, and gases like nitrate, sulfate, ammonia, etc. The sources of these particles include both natural (dust storms, volcanoes, land fires) and manmade (vehicular traffic, construction work, central heating). Sometimes during a specific time period, their concentrations suddenly increased and affect the daily life routine of the inhabitants. The public must be prior informed about these hike span time and the government should try to reduce these concentrations through extra efforts like odd–even vehicle formula applied in Delhi [3]. We can conclude from the above discussion that short-term and long-term forecasting is a prime demand in urban planning and development areas.

Air forecasting is a wide area and a large number of methods are scattered in the literature on this. These methods can be further divided into four categories:

1. Deterministic Methods: These methods do not require a large amount of data as these involve the numerical solution of differential equations. This process needs a precise knowledge of differential calculus as well as chemical terms like pollutants sources, emitted number of particles, their components, etc. [4]. The results obtained through these methods are quite uncertain and cannot be suitable for planning and warning [5, 6].
2. Statistical Methods: These models provide more accurate predictions in comparison to deterministic methods as these are based on a linear relationship between pollutants and different variables. Box models [7] and regression models [8] are some of the examples of statistical type air forecasting models. Later on, neural network models [9, 10] and least square support vector machine [11] are used for forecasting purposes as these have very few restrictions on the input data. The main drawback of these methods is that the same method could not be used at different locations.
3. Chemical Transport Method: The methods which rely on the chemical properties of pollutants lie under this category but not became popular due to their poor performance.
4. Hybrid Methods: Methods based on the hybridization concept of two different models are kept under this category. Computational-based techniques like extreme learning machine, wavelet transform based neural networks, and data decomposition techniques like empirical mode decomposition and variable mode decomposition are some of the examples. Further several metaheuristic algorithms including Particle swarm optimizer, Grey wolf optimizer, Harris Hawk optimization also joined this category [11–13]. These models are helpful to users due to easy implementation and better efficiency.

Inspiring from these approaches, we propose a supervised architecture based on Feed-Forward Neural Network (FFNN), for tuning the weights of FFNN we employ our recently proposed Intelligent Crow Search Algorithm (ICSA).

In the next section, development of ICSA will be discussed. Section 3 presents the details of the proposed forecasting model. In Sect. 4, results are presented and in last the work of the paper is concluded in the conclusion section.

2 Intelligent Crow Search Algorithm

Crow Search Algorithm (CSA) has been proposed by Askharzadeh [14]. This algorithm is inspired by the behavior of crows which can be considered as the most common but also as the most intelligent bird among its category. According to research [15], the brain size is bigger in comparison to other birds in the same category. It proves its efficiency in mirror test, tool making [16]. Crow steals the food of another crow and tries to hide it from other crows in their flock, in this process, they also make fool others by changing their position. This algorithm mimics the hiding and stealing food behavior of crows and attracts the attention of researchers due to limited parameters and simple structure.

CSA has been used to solve a wide variety of real and engineering problems. In [17], the authors proposed a chaotic crow search algorithm and then optimize a feature selection problem as an application. An improved version of crow search has been applied to solve energy problems in [18]. An optimized version of crow search was introduced in [19] and further used in the diagnosis of Parkinson's disease. A modified crow search algorithm is proposed and then applied in the extraction and prediction of usable features of a hierarchical model in [20]. Similarly, CSA has also been used in the solutions of high-dimensional problems [21] and fractional optimization problems [22]. An intelligent CSA was presented in [23] and two different problems: Model order reduction and structural design problem. In this paper, two theories were chosen to improve the results of CSA, which are

- (1) First, a concept of learning based on opposition has been applied in the initialization of the optimization process where half of the number of crows generated randomly and another half portion generated opposite to them. The concept of opposition can be defined as

Definition: If $x = x_i (i = 1, \dots, r)$ is a point in an R-dimensional space, x_i is a real number for $i \in \{1, 2, \dots, r\}$, and bounded in $[p, q]$ then the set of opposite points of x is defined as

$$\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_r) \text{ or } \bar{z}_i = [a_i + b_i - z_i]. \quad (1)$$

- (2) As the second theory, we applied a cosine function based acceleration factor which also maintains a balance between intensification and diversification phases in optimization. The gradient of cosine factor is high which is helpful in diversification as it can approach a bigger area in solution finding, while in

the case of low gradient the chances of local minima entrapping reduce in a smaller area. We can define it as

$$acc.f. = \cos^2\left(\frac{\omega}{2}\right) \quad (2)$$

$$\omega = \frac{\pi(I_{current})}{I_{Max}} \quad (3)$$

We have assumed that a crow always lives in a group (flock), it remembers the position of its hidden food, tries to steal the food of another crow, and protective about its food.

The position of hidden food be M_r^p , which can also consider as the best hidden food position of rth crow at pth iteration, where r can be $\{1, \dots, N\}$, i.e., number of flocks. In the initial step, they generate half of their position randomly and the other half population is generated by following opposition theory defined in Eq. (1). When a crow is followed by another, then either he knows about the following crow or he does not know about it, if he knows that another crow is following him, he wants to fool him by swiftly changing its position.

These two cases can be reflected mathematically as

$$F_r^{p+1} = \begin{cases} F_r^p + (acc.f.) R_r f_r (M_r^p - F_r^p) & \text{if } R_r \geq AP^{r,p} \\ \text{a random number} & \text{otherwise} \end{cases} \quad (4)$$

where R_r is a random number evenly distributed between 0 and 1, f_r is flight length of rth crow, and term $AP^{r,p}$ represents awareness probability of rth crow at pth iteration. This factor creates a balance between the intensification and diversification process. The final equation by which crow updates its position in each iteration can be given as

$$F_r^{p+1} = \begin{cases} F_r^{p+1}, & fn(F_r^{p+1}) \text{ is better than } fn(M_r^p) \\ F_r^p & \text{otherwise} \end{cases} \quad (5)$$

3 Construction of the Forecasting Architecture Model

We have chosen the feed-forward neural network for the construction of our time series-based forecasting architecture. There are three layers in this type of neural network known as input layer, output layer, and hidden layers. These layers are interconnected with each other through weights. To find the values of these biases and weights, we minimize the cost function given by

$$W = \min_{w_i} \left(\sum_{i=1}^N y_i w_i - x_t \right)^2$$

where x_t represents the required output, which is known at time t, y_t is input, and w_i denotes the weights in the network.

For optimal tuning of weights and to minimize the error between the actual values and predicted values we can use different optimization algorithms. Here we construct a time-series-based forecasting method in which day 1 to day 4 values are considered as input and day 5 value is taken as the output value. This process is followed through the whole data and creates a rolling model. A rolling forecast is an add/drop process for predicting the future over a set period of time. In this work, we chose a rolling window of four days for predicting the PM10 concentration (Figs. 1 and 2).

4 Results

As explained in the previous section, neural network weight training has been executed by evolutionary algorithms, and comparison on the basis of error indices is done with ICSA. For comparing the performance of ICSA, we have chosen four frontier algorithms that come in nature-inspired optimizers category.

They are as follows.

- Moth Flame Optimization [24],
- Sine Cosine Algorithm [25],
- Harris Hawk Optimization [26], and
- Crow Search Algorithm.

After training of the neural networks, unknown samples are taken and results are obtained in terms of error indices such as Mean Square Error (MSE), Mean absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). Following conclusions can be driven from this analysis.

1. Unknown Samples (US) are taken as per Fig. 3, the reason we address those unknown samples is that these samples are not involved in training or validation phases.
2. We observe from Table 1 that the modification developed in ICSA helps search agents to jump the local minima stages and further these modifications help to converge just near to the accurate values. Table 1 shows the simulated results.
3. Table 2 shows diverse error indices on the basis of these indices one can easily conclude that ICSA outperforms all other algorithms and yields fruitful results in terms of accuracy. From Table 2, we can observe that the values of MAPE are optimal for ICSA (2.06) however, SCA-tuned FFNN gives pessimistic results as the recorded MAPEs are highest.

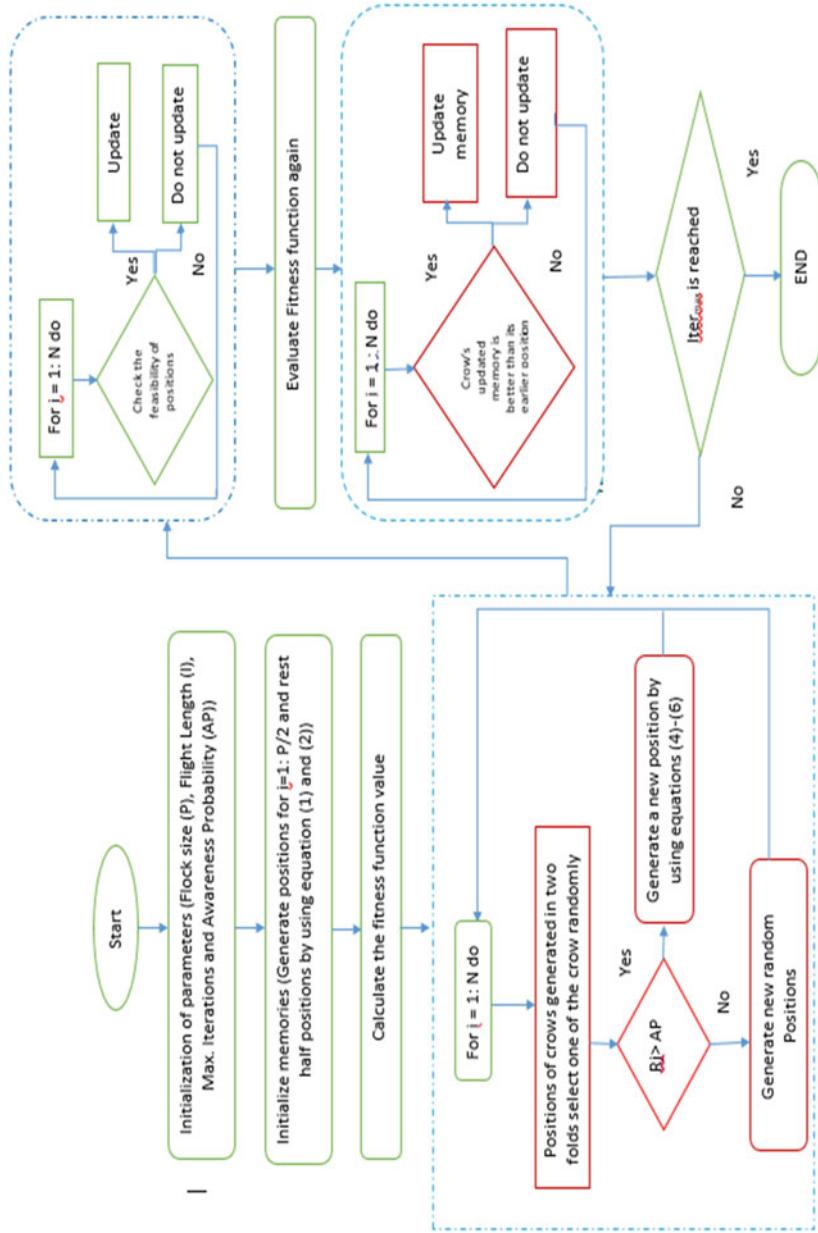
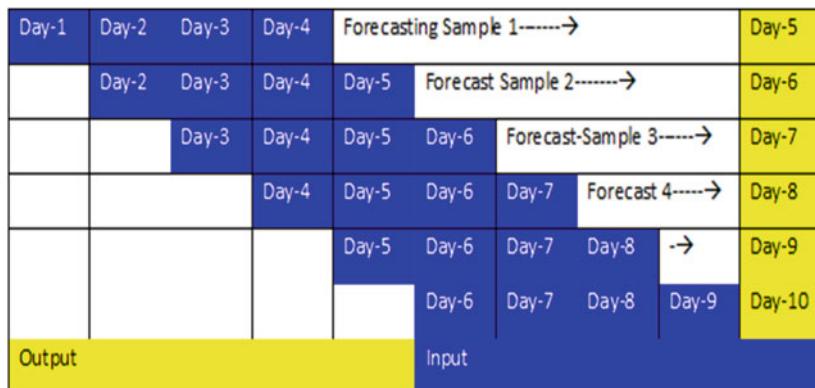
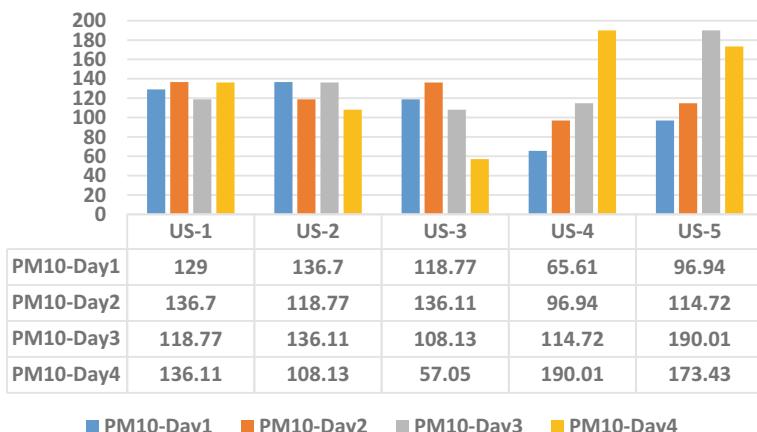


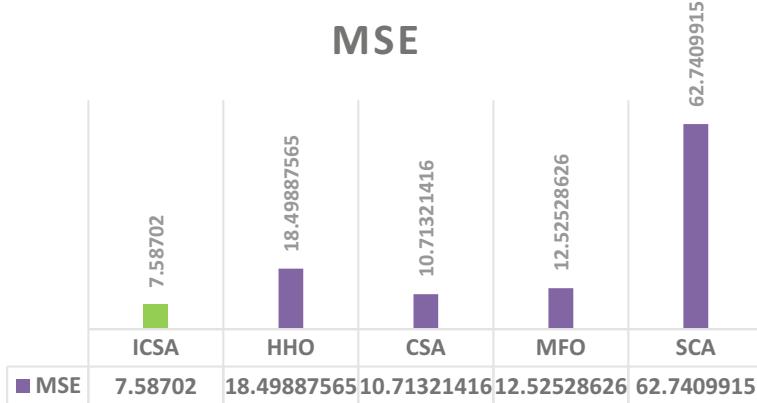
Fig. 1 Flowchart of ICSA [9]

**Fig. 2** Construction of forecasting architecture model**Fig. 3** Unknown samples for testing**Table 1** Forecasted results of PM₁₀ for Jaipur city

Sample	Target	ICSA	HHO	CSA	MFO	SCA
US-1	108.13	105.9	105.1061	105.423	104.8023	105.1909
US-2	57.05	58.69	58.77703	60.8598	54.37613	53.01523
US-3	65.61	64.56	59.93571	69.5643	72.0696	77.24561
US-4	173.43	172.69	177.5766	177.3648	174.9831	185.3799
US-5	158.59	153.24	153.0244	159.3675	159.1048	161.8458

Table 2 Comparative analysis on the basis of error indices

Error Indices	ICSA	HHO	CSA	MFO	SCA
RMSE	2.754455	4.301032	3.273105	3.539108	7.920921
MSE	7.58702	18.49888	10.71321	12.52529	62.74099
MAPE	2.067507	4.074534	3.593503	3.76598	7.293642

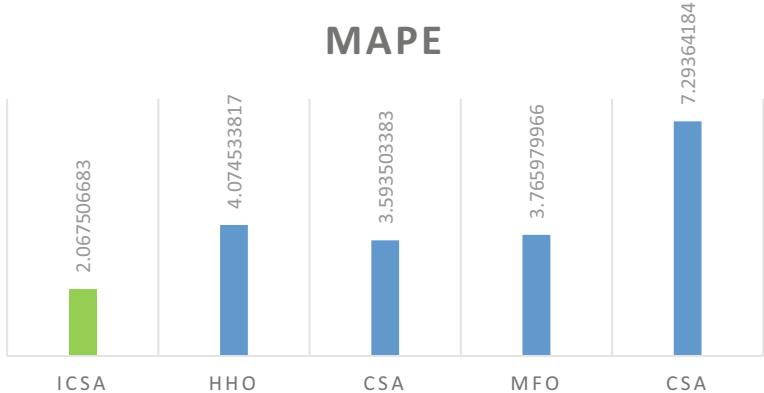
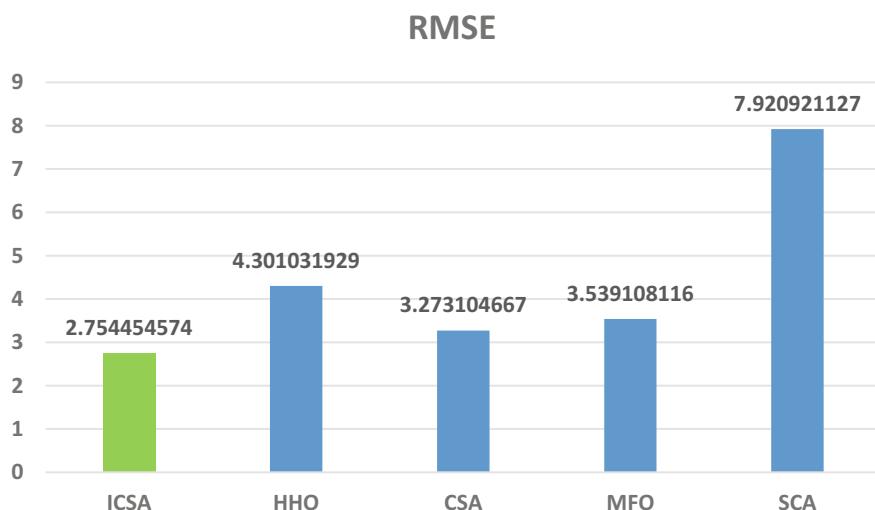
**Fig. 4** Comparison of MSE

Further, by inspecting the values of RMSE and MSE, it can be easily concluded that ICSA yields better results as compared to other competitors and we can conclude that SCA is not a proper choice to construct supervised architecture for the prediction of PM10 for this particular case. Further, for a better understanding, these results are depicted in terms of figures also.

By judging this figure also, we can easily conclude that these algorithms achieve acceptable MAPE and other error indices for forecasting the pollutants. Results of ICSA have been shown distinctly in these Figs. 4, 5, and 6. Further, the decisive conclusions arrived from this study are presented in the next section.

5 Conclusion

Pollution forecast is a challenging problem and considers as a fulcrum of city planning especially in urban areas. This work focuses on the development of supervised architecture based on FFNN and evolutionary algorithms. In view of this, recently developed algorithm by authors has been employed to train the neural network by identifying appropriate weights. We observe that ICSA yields better results as compared with HHO, CSA, MFO, and SCA algorithms. The decisive evaluation of this structure has been carried out by the calculation of several error indices such

**Fig. 5** Comparison of MAPE**Fig. 6** Comparison of RMSE

as MAPE, MSE, and RMSE. We observe these values are optimal for ICSA. In our future work, we will develop a supervised routine for predicting other pollutants also.

Acknowledgements The authors are thankful for the full financial support from the CRS, RTU (ATU), TEQIP-III of Rajasthan Technical University, Kota, Rajasthan, India. (Project Sanction No. TEQIP-III/RTU (ATU)/CRS/2019-20/51).

References

1. Scungio, Mauro et al., Lung cancer risk assessment at receptor site of a waste-to-energy plant. *Waste Manag.* **56**, 207–215 (2016)
2. Sarkar, Sayantan et al., Chemical speciation of respirable suspended particulate matter during a major firework festival in India. *J. Hazard. Mater.* **184.1–3**, 321–330 (2010)
3. S.K. Sharma et al., Study on ambient air quality of megacity Delhi, India during odd–even strategy. *Mapan* **32**(2), 155–165 (2017)
4. Hrust, Lovro et al., Neural network forecasting of air pollutants hourly concentrations using optimised temporal averages of meteorological variables and pollutant concentrations. *Atmosph. Environ.* **43**.35, 5588–5596 (2009)
5. P. Goyal, Andy T. Chan, Neeru Jaiswal, Statistical models for the prediction of respirable suspended particulate matter in urban cities. *Atmosph. Environ.* **40**.11, 2068–2077 (2006)
6. A.J. Jakeman, R.W. Simpson, J.A. Taylor, Modeling distributions of air pollutant concentrations—III. The hybrid deterministic-statistical distribution approach. *Atmosph. Environ.* (1967) **22**.1, 163–174 (1988)
7. D.R. Middleton, A new box model to forecast urban air quality: BOXURB. *Environ. Monit. Assess.* **52**(1–2), 315–335 (1998)
8. Shi, Ji Ping, Roy M. Harrison, Regression modelling of hourly NO_x and NO₂ concentrations in urban air in London. *Atmosph. Environ.* **31**(24), 4081–4094 (1997)
9. Bai, Yun, et al., An ensemble long short-term memory neural network for hourly PM_{2.5} concentration forecasting, *Chemosphere* **222**, 286–294 (2019)
10. Y. Hao, C. Tian, The study and application of a novel hybrid system for air quality early-warning. *Appl. Soft Comput.* **74**, 729–746 (2019)
11. Wang, Deyun, et al., A novel hybrid model for air quality index forecasting based on two-phase decomposition technique and modified extreme learning machine. *Sci. Total Environ.* **580**, 719–733 (2017)
12. Zhou, Qingping, et al., A hybrid model for PM_{2.5} forecasting based on ensemble empirical mode decomposition and a general regression neural network. *Sci. Total Environ.* **496**, 264–274 (2014)
13. Saxena, Akash, Shalini Shekhawat, Ambient air quality classification by grey wolf optimizer based support vector machine. *J. Environ. Public Health* **2017** **11** (2017)
14. A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. *Comput. Struct.* **169**, 1–12 (2016)
15. Rincon, Paul, Science/naturel crows and jays top bird IQ scale. BBC News (2005)
16. Prior, Helmut, Ariane Schwarz, Onur Güntürkün, Mirror-induced behavior in the magpie (*Pica pica*): evidence of self-recognition. *PLoS Biol.* **6**.8, e202 (2008)
17. Sayed, Gehad Ismail, Aboul Ella Hassanien, Ahmad Taher Azar, Feature selection via a novel chaotic crow search algorithm. *Neural Comput. Applicat.* **31**.1, 171–188 (2019)
18. Díaz, Primitivo, et al., An improved crow search algorithm applied to energy problems. *Energies* **11**.3, 571 (2018)
19. Gupta, Deepak, et al., Improved diagnosis of Parkinson's disease using optimized crow search algorithm. *Comput. Elect. Eng.* **68**, 412–424 (2018)
20. Gupta, Deepak, et al., Usability feature extraction using modified crow search algorithm: a novel approach. *Neural Comput. Appl.* 1–11 (2018)
21. M. Jain, A. Rani, V. Singh, An improved Crow Search Algorithm for high-dimensional problems. *J. Intell. Fuzzy Syst.* **33**(6), 3597–3614 (2017)
22. Rizk-Allah, Rizk M., Aboul Ella Hassanien, Siddhartha Bhattacharyya, Chaotic crow search algorithm for fractional optimization problems. *Appl. Soft Comput.* **71**, 1161–1175 (2018)
23. S. Shekhawat, A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition-based learning. *ISA Trans.* **99**, 210–230 (2020)
24. S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl.-Based Syst.* **89**, 228–249 (2015)

25. S. Mirjalili, SCA: a sine cosine algorithm for solving optimization problems. *Knowl.-Based Syst.* **96**, 120–133 (2016)
26. Heidari, Ali Asghar, et al., Harris Hawks optimization: algorithm and applications. *Future Gen. Comput. Syst.* **97**, 849–872 (2019)

A Hybrid Fusion-Based Algorithm for Underwater Image Enhancement Using Fog Aware Density Evaluator and Mean Saturation



Rosalind Margaret Paulson, Sruthi Gopalakrishnan, Sruthi Mahendiran, Varghese Paul Srambical, and Neethu Radha Gopan

Abstract Underwater images are degraded mainly due to scattering and absorption effects but are key in oceanographic studies and research. Therefore, we need to develop methods that generate visually pleasing images and retain the original information. In this paper, we propose a method that chooses between Multiscale Fusion, Edge Preserving Decomposition-Based Haze Removal Algorithm or a combination of both. The algorithm that is to be used in an image is based on mean saturation value and fog density using Fog Aware Density Evaluator (FADE). The resulting image retains the natural color distribution, is dehazed and enhanced. The proposed algorithm doesn't require prior hardware usage or prerequisite knowledge of the underwater environment. The proposed algorithm performs considerably well when compared to previous approaches against various image quality metrics such as UIQM, PCQI, PIQE, BRISQUE and Average Gradient.

Keywords Underwater image enhancement · Multiscale fusion · Dehazing · Edge preserving decomposition-based single image haze removal · Fog aware density evaluator · Gray world algorithm

1 Introduction

Underwater images show a decrease in visibility effects and wavelength-dependent color distortion due to absorption and scattering effects. Images taken underwater are almost completely dependent on environmental conditions. Underwater image restoration is considered as a distance-dependent degradation problem based on the degradation model proposed by Koschmeider in [1]. The single image fusion-based approach proposed by Ancuti in [2] gives good results visually but has lower values in quality metrics compared to other approaches. Galdran, in [3] proposed that assuming

R. M. Paulson (✉) · S. Gopalakrishnan · S. Mahendiran · V. P. Srambical · N. R. Gopan
Department of Electronics and Communication Engineering, Rajagiri School of Engineering and Technology, Kochi, Kerala, India
e-mail: neethurg@rajagritech.edu.in

the transmission map to be constant over a patch is unrealistic and that the usage of a Guided Image Filter (GIF) efficiently captures the finer details of a degraded image and incorporates it into the estimated transmission map. However, Li and Zheng in [4] stated that GIFs might exhibit halo artifacts near some edges which is due to unwanted smoothing of edges. This is not seen in Weighted Least Square (WLS) filters. This is attributed to the Lagrangian factor being adaptive in the WLS filter and constant in the GIF. Thus, edge-aware weighting is added to the GIF to form a Weighted Guided Image Filter (WGIF) that is used in the proposed paper. This assumption is effective since the haze in underwater images is removed considerably. The approach, however, confers artificiality to some images when the image lacks a considerable amount of haze. Using a processing technique prior to the Edge preserving approach like the Gray World algorithm by Ebner and Marc in [5] makes the image have an acceptable perceptual quality but makes it underperform in a qualitative evaluation. Subjecting the images to the algorithm in [2] and then [4] performs well in the qualitative evaluation but increases the artificiality in most images. This paper proposes an approach that unites the acceptable perceptual quality obtained in the former and the better qualitative metric values obtained in the latter. The classification of the images in order to decide the processing approach to be used was done on the basis of the density value obtained from using the FADE algorithm by Choi et al. [6] which predicts the visibility of a foggy scene in a single image without reference to a corresponding fog-free image. It only makes use of measurable deviations from statistical regularities observed in natural foggy and fog-free images. In addition to this, it uses the mean saturation value. The resulting images have better global contrast, perform better in qualitative evaluation and are also visually appealing.

2 Proposed Method

Underwater images show loss of detail and a large amount of haze. This is a drawback in oceanographic studies and research applications. Our emphasis is on images that have been severely degraded due to scattering, attenuation and those with a wide range of objects rather than single object images. This is necessary to prove the efficiency of the algorithm. This algorithm selects between multiscale fusion, edge preserving decomposition-based haze removal algorithm or a combination of both depending on the value of fog density using Fog Aware Density Evaluator (FADE) and mean saturation. This ensures that the natural colors of the image are retained. The processed image shows no haze, high accuracy, good recovery of lost details, reduction in color cast and improvement in color contrast. The method does not make use of any prior (Fig. 1).

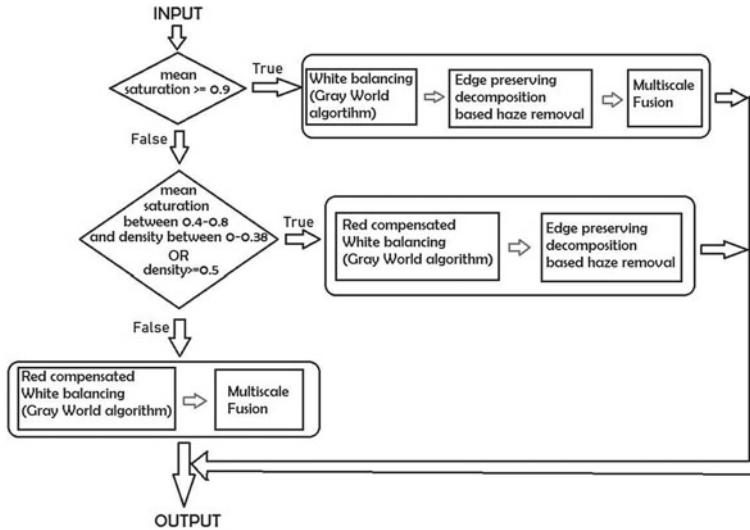


Fig. 1 Proposed algorithm

2.1 Multiscale Fusion

Multiscale Fusion builds on the blending of two images which are introduced to enhance the color contrast and the edge sharpness of the white-balanced image. These are the gamma-corrected and the sharpened image, respectively. These two input images as well as their associated weight maps are defined to promote the transfer of edges and color contrast to the output image [2]. Equation for sharpening is given by

$$S = (I + N(I - G * I))/2 \quad (1)$$

where N is the linear normalization operator, $G * I$ denotes the Gaussian filtered version of the image and S stands for the sharpened image. Equation for Gamma correction is given by

$$V_{\text{out}} = V_{\text{in}}^{\gamma} \quad (2)$$

where V_{out} is the output luminance value and V_{in} is the input luminance value.

Following this, the weight maps, namely Laplacian, saturation and saliency are found which are then fused using the concept of Laplacian pyramid and the N levels of the Laplacian pyramid can be represented as

$$\begin{aligned}
I(x) &= I(x) - G_1\{I(x)\} + G_1\{I(x)\} \triangleq L_1\{I(x)\} + G_1\{I(x)\} \\
&= L_1\{I(x)\} + G_1\{I(x)\} - G_2\{I(x)\} + G_2\{I(x)\} \\
&= L_1\{I(x)\} + L_2\{I(x)\} + G_2\{I(x)\} \\
&= \dots \\
&= \sum_{l=1}^n L_l\{I(x)\}
\end{aligned} \tag{3}$$

In this equation, L_l and G_l represent the l th level of the Laplacian and Gaussian pyramids, respectively. Each source input I_k is decomposed into a Laplacian pyramid while the normalized weight maps \overline{W}_k are decomposed using a Gaussian pyramid. Both pyramids have the same number of levels and the mixing of the Laplacian inputs with the Gaussian normalized weights is performed independently at each level l :

$$R_l(x) = \sum_{k=1}^K G_l(\overline{W}_k(x))L_l(I_k(x)) \tag{4}$$

where l denotes the pyramid levels and k refers to the number of input images. The dehazed output is obtained by taking the sum of the fused contributions of all levels after appropriate upsampling (Fig. 2).

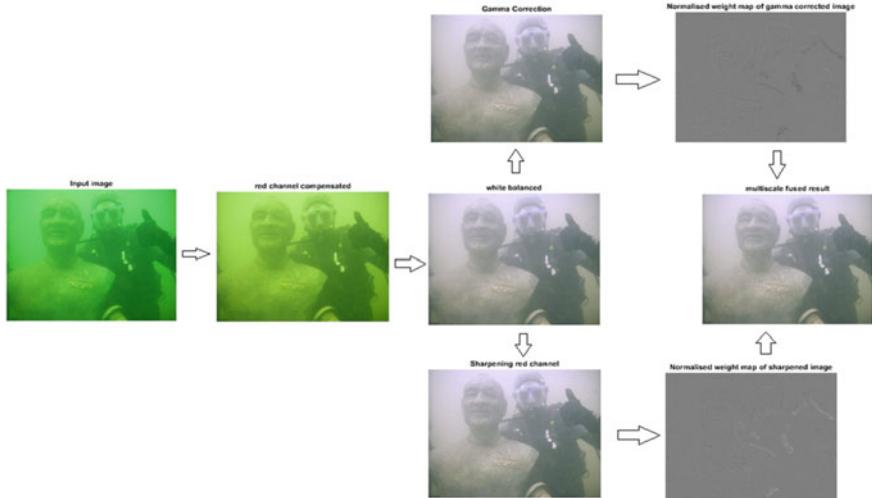


Fig. 2 Steps involved in Multiscale Fusion

2.2 Gray World White Balancing Algorithm

The Gray World Algorithm assumes that the average reflectance in a scene is achromatic [2, 5]. The illuminant color distribution is found by averaging each channel. The gray world assumption holds if and only if we have a good distribution of colors in the scene. Performance analysis on the gray world algorithm shows that it performs well in extremely deteriorated underwater images but there is an occurrence of red artifacts due to the small mean value of the red channel.

$$\text{Gray} = R_{\text{avg}} + G_{\text{avg}} + B_{\text{avg}} \quad (5)$$

where R_{avg} , G_{avg} and B_{avg} represent the mean values of red, blue and green channels.

2.3 Red-Compensated White Balancing Algorithm+Gray World Algorithm

The main drawback while using the gray world algorithm is the presence of red artifacts. This is mainly due to the small mean value of the red channel, leading to overcompensation in regions where red is present. In order to overcome this, the loss of the red channel is compensated. This is done by adding a portion of the green channel, which has opponent information, in regions where attenuation of the red channel was maximum [1]. The compensation incorporated must be the difference between the mean green value and the mean red value. This is necessary since the assumption in the gray world algorithm is that the mean values of channels before attenuation are similar. Once red compensation has been done, a normal gray world algorithm is applied. The equation for red compensation is given by

$$I_{rc}(x) = I_r(x) + \alpha \cdot (\overline{I_g} - \overline{I_r}) \cdot (1 - I_r(x)) \cdot (I_g(x)) \quad (6)$$

where $\alpha = 1$, I_r , I_g represent the red and green color channels of image I, each channel being in the interval $[0, 1]$, after normalization by the upper limit of their dynamic range. $\overline{I_r}$ and $\overline{I_g}$ denote the mean values of I_r and I_g .

2.4 Edge Preserving Decomposition-Based Single Image Haze Removal Algorithm

The approach in [4] estimates the transmission map of a hazy image in order to develop a single image haze removal algorithm based on Koschmieder's model [1]. In this algorithm, the dark channel of the hazy image only has the function of reducing the variation in $(A_m - J_d^Z(p))$. This is important because this ensures that the

Weighted Guided Filter (WGIF) is able to decompose the simplified dark channel of the hazy image into a base layer and a detail layer. The transmission map is essentially the base layer, subject to some refinement by the WGIF. Estimating transmission map results in a haze-free image. The equation for estimating transmission map is given by

$$t^*(p) = 1 - \frac{\psi_p^*}{A_m} \quad (7)$$

where ψ_p^* represents the optimal solution and A_m is the minimum value of atmospheric light in all the three channels (R, G, B). The equation for recovering scene radiance once transmission map is found is given by

$$Z_c(p) = \frac{1}{t^*(p)}(\hat{X}_c(p) - A_c) + A_c + \frac{1}{t^* p} e_c(p) \quad (8)$$

$$X_c(p) = \hat{X}_c(p) + e_c(p) \quad (9)$$

where $Z_c(p)$ is the scene radiance, $t^*(p)$ is the estimated transmission map, $X_c(p)$ is the input image including noise which is denoted by $e_c(p)$ and A_c is the atmospheric light (Fig. 3).

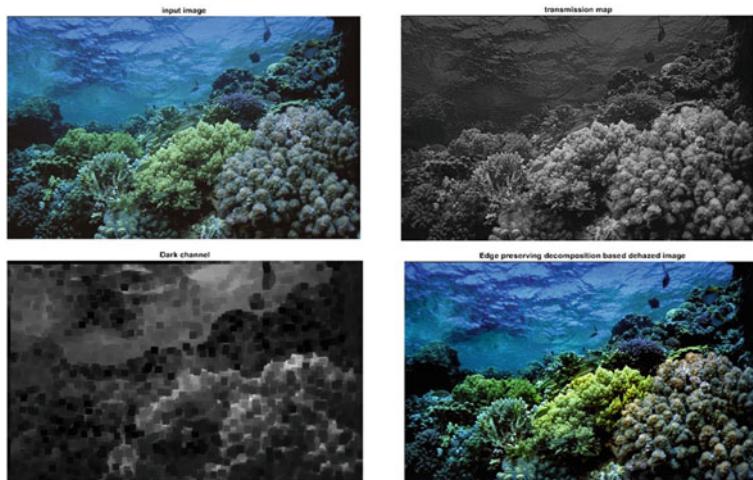


Fig. 3 Steps involved in edge preserving decomposition-based single image haze removal

2.5 Fog Aware Density Evaluator

A Fog Aware Density Evaluator (FADE) detects the visibility of a foggy scene without using a fog-free image as a reference. It does not depend on salient objects, requires estimation of a depth-dependent transmission map and human-rated training [6]. The prediction model is based on Natural Scene Statistics (NSS) and fog-aware statistical features. Deviations from scene statistical features are measured like that of the Mahalanobis distance where D_f represents the distance between the multivariate Gaussian function applied to a test foggy image against a database of foggy images. This helps in establishing a foggy level. When the same distance is measured for a fog-free image, it gives a fog-free level D_{ff} . Density value is calculated as the ratio of foggy level to that of a fog-free level and the equation is given by

$$D = \frac{D_f}{D_{ff} + 1} \quad (10)$$

FADE determines the perceptual fog density of an entire image and the local fog density of patches. The result of fog density is measured and correlated to that of visual judgements made by human beings on a large database. An underwater image is taken and given as an input to calculate the fog density using FADE [6] and the mean saturation (image is converted from RGB color model to HSV color model). Depending upon this, various selections are made.

For really high mean saturation values (i.e., mean saturation ≥ 0.9), the image is first white-balanced using the gray world algorithm followed by the edge preserving decomposition-based method. This results in images looking artificial due to overcompensation. Hence, multiscale fusion is done [1] with values of $\lambda = 65536$, $\zeta_1 = 60$, $\zeta_2 = 15$ and $\gamma = 1$ (see Fig. 5).

For images that have moderate mean saturation value (0.4-0.8) and very low fog density (0-0.38) or for heavily hazed images with density value ≥ 0.5 , initially red-compensated white balancing is done on the image. This image is then dehazed using the Edge Preserving Decomposition-based Haze Removal technique, for values of $\lambda = 65536$, $\zeta_1 = 60$ and $\zeta_2 = 15$ (see Fig. 6).

When mean saturation and fog density values do not belong to any of the previously mentioned values, the input image is subject to red-compensated white balancing initially and then multiscale fusion is performed for $\gamma = 1$ (see Fig. 7).

3 Results and Discussion

We compare our algorithm and [2] with the help of quality metrics like UIQM [7], PCQI [8], PIQE [9], BRISQUE [10] and Average Gradient.

PCQI is a local patch-based objective reference image quality metric that decomposes an image patch into three parameters—mean intensity, signal strength and



Fig. 4 Input images

signal structure. Their perceptual distortions are then evaluated. UIQM is a referenceless underwater image quality measure addressing three underwater image attributes: Colorfulness (UICM), Sharpness (UISM) and Contrast (UIConM), which evaluate the images based on properties of Human Visual Systems (HVSs). PIQE calculates the no-reference image quality score for an image using a perception-based image quality evaluator. BRISQUE calculates the no-reference image quality score for an image and compares it to a default model computed from images of natural scenes with similar distortions. Average Gradient is used to measure the clarity of an image. All images have been taken from the SUN database [11], Reefbase [12], MATLAB Central File Exchange [13], the companion repository of [14] and from [15]. About 100 images were processed using the proposed algorithm of which 90 images performed well. A common observation in the other images was that they consist of very high pixel values, almost white against a predominantly dark blue background. These images are characterized by unnatural brightening.

The input images in Fig. 4 are processed and their evaluation is made in Table 1. The image dataset can be classified into four categories: (a) Shipwreck, (b) Divers, (c) Aquatic flora and fauna and (d) Rocks. On comparing the two algorithms, the general trend is the marginal increase in the values of PCQI, UIQM and PIQE. The mandatory decrease in the BRISQUE value, an indicator of better perceptual quality, is also seen quite markedly in the case of images of the ocean. A bucking in the trend is observed in the case of diver images, where the BRISQUE value increases, causing image degradation. A steep increase is seen in the value of the average gradient in our method, which leads to the conclusion that our images have more clarity. Overall,

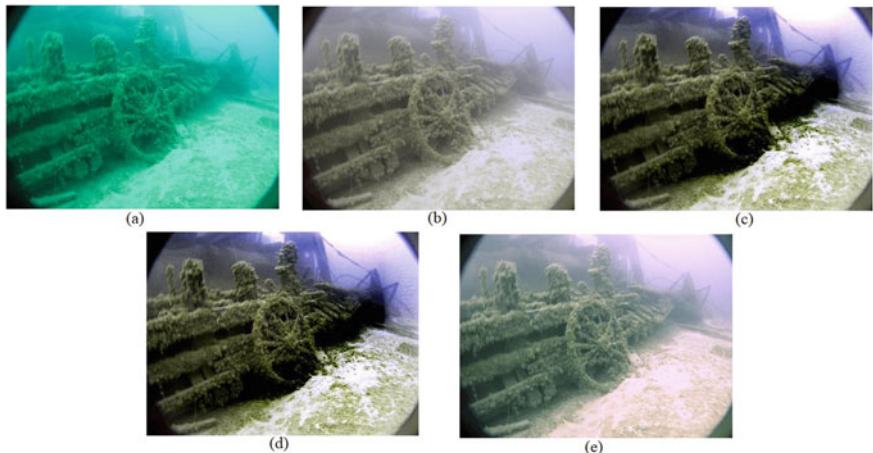


Fig. 5 An image with mean saturation value ≥ 0.9 **a** Input image; **b** Image obtained after white balancing using gray world algorithm; **c** Image obtained after edge algorithm; **d** Proposed algorithm; **e** An cuti [2]

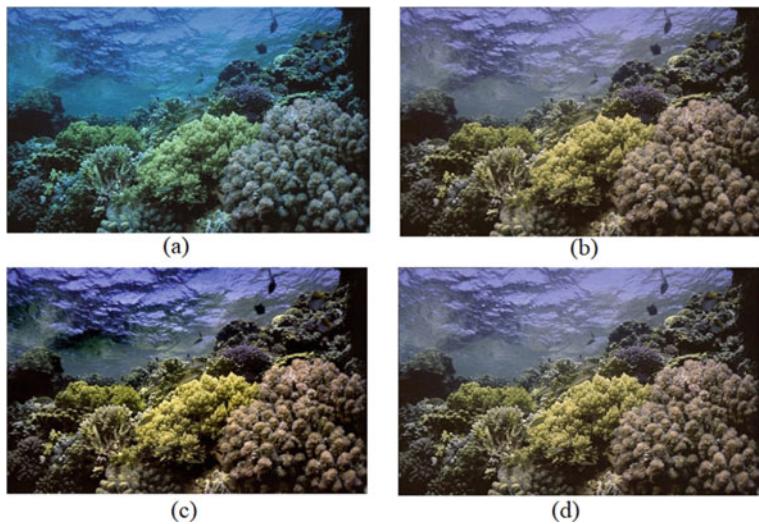


Fig. 6 An image with mean saturation value range of 0.4–0.8 and density values either ranging from 0 to 0.3 or > 0.5 . **a** Input image; **b** Image obtained after red-compensated white balancing; **c** Proposed algorithm; **d** An cuti [2]

Table 1 QUANTITATIVE EVALUATION OF IMAGES

Ancuti [2]		Proposed Algorithm								
	PCQI	UIQM	BRISQUE	PIQE	Average Gradient	PCQI	UIQM	BRISQUE	PIQE	Average Gradient
Coral Reef	1.043	4.9128	12.5549	28.0294	76.3986	1.0587	5.0435	9.15	22.4671	79.3456
Shipwreck	0.9867	4.1465	24.5377	17.3435	30.0442	1.1171	12.4854	22.6466	25.28	53.4907
Rock	0.9967	3.977	33.6048	15.1441	19.291	1.0182	6.1666	30.8629	18.2744	31.5575
Ocean	0.9715	2.3206	21.6357	32.242	17.7831	1	2.6773	0.778	43.978	31.8992
Divers1	0.9144	3.6067	10.3468	8.2825	14.269	1.0288	4.4729	23.5224	20.1081	29.2869
Corall	1.0564	5.3823	29.6964	16.0605	73.725	1.1055	4.5873	29.0618	22.2001	84.9617
Coral2	0.9401	3.2803	33.6041	16.1103	9.1236	1.0184	4.9315	31.604	19.0473	23.9433
Turtle	0.9011	3.4195	27.8084	38.2383	20.19	1.1478	4.5051	19.276	30.7515	59.8394
Hydrophyte	0.9321	3.2195	44.3523	42.4163	15.153	1.2456	4.7201	30.515	44.4219	48.2929
Divers2	0.9261	1.5623	38.1248	31.4366	8.236	1.0596	3.3779	25.266	23.2858	29.1743

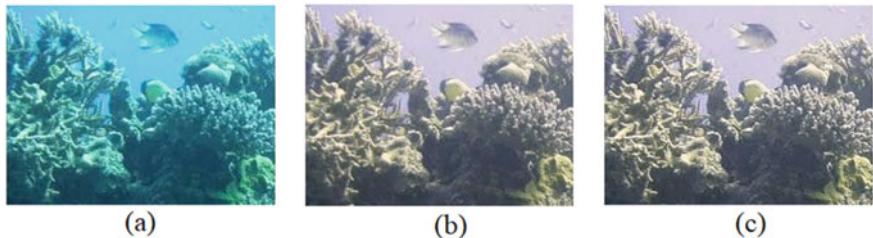


Fig. 7 An image with mean saturation and fog density value that does not satisfy either of the conditions. This image is subjected to white balancing and multiscale fusion alone. (a) Input image (b) Image obtained after red compensated white balancing (c) Output image [2]

we conclude that the resulting images are of good perceptual quality with significant enhancement of image structure details, color and contrast.

4 Conclusion

It can be concluded that the proposed algorithm performed better than existing techniques in the enhancement of a wide variety of underwater images which is evident from the value of the quality metrics used. The main limitations are the emergence of whitish spots at some images in their background, overcoming which can be the basis of any future work.

References

1. H. Koschmieder, Theorie der horizontalen sichtweite. *Beitrage Phys. Freien Atmos.* **12**, 171–181 (1924)
2. C.O. Ancuti, C. Ancuti, C. De Vleeschouwer, P. Bekaert, Color balance and fusion for underwater image enhancement. *IEEE Trans. Image Process.* **27**(1), 379–393 (2018). <https://doi.org/10.1109/TIP.2017.2759252>. Jan
3. A. Galdran, D. Pardo, A. Picon, A. Alvarez-Gila, Automatic red-channel underwater image restoration. *J. Vis. Commun. Image Representation.* **26** (2014). <https://doi.org/10.1016/j.jvcir.2014.11.006>
4. Z. Li, J. Zheng, Edge-preserving decomposition-based single image haze removal. *IEEE Trans. Image Process.* **24**(12), 5432–5441 (2015). <https://doi.org/10.1109/TIP.2015.2482903>. Dec
5. M. Ebner, *The Gray World Assumption* (Wiley, Color Constancy. Chichester, West Sussex, 2007)
6. L.K. Choi, J. You, A.C. Bovik, Referenceless prediction of perceptual fog density and perceptual image defogging. *IEEE Trans Image Process.* **24**(11), 3888–3901 (2015). Nov

7. K. Panetta, C. Gao, S. Agaian, Human-visual-system-inspired underwater image quality, measures. *IEEE J. Oceanic Eng.* **41**(3), 541–551 (2016). <https://doi.org/10.1109/JOE.2015.2469915>. July
8. S. Wang, K. Ma, H. Yeganeh, Z. Wang, W. Lin, A patch-structure representation method for quality assessment of contrast changed images. *IEEE Signal Process. Lett.* **22**(12), 2387–2390 (2015). <https://doi.org/10.1109/LSP.2015.2487369>. Dec.
9. N. Venkatanath, D. Praneeth, BhM Chandrasekhar, S.S. Channappayya, S.S. Medasani, Blind image quality evaluation using perception based features, in *Proceedings of the 21st National Conference on Communications (NCC)* (IEEE, Piscataway, NJ, 2015)
10. A. Mittal, A.K. Moorthy, A.C. Bovik, No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.* **21**(12), 4695–4708 (2012)
11. J. Xiao, J. Hays, K. Ehinger, A. Oliva, A. Torralba, SUN database: large-scale scene recognition from abbey to zoo, in *IEEE Conference on Computer Vision and Pattern Recognition* (2010)
12. Reefbase.org. [http://www.reefbase.org/resource\\$_center/photos.aspx?stress=BL](http://www.reefbase.org/resource$_center/photos.aspx?stress=BL)
13. T. Huang . Underwater Images. MATLAB Central File Exchange (2020). <https://www.mathworks.com/matlab-central/fileexchange/51082-underwater-images>
14. L.K. Choi, J. You, A.C. Bovik, LIVE image defogging database (2015). [http://live.ece.utexas.edu/research/fog/fade\\$_defade.html](http://live.ece.utexas.edu/research/fog/fade$_defade.html)
15. Coral Reef Puerto Rico. <https://web.whoi.edu/singh/underwater-imaging/datasets/coral-reef-puerto-rico/>

Application of Hybridized Whale Optimization for Protein Structure Prediction



Akash Saxena, Shalini Shekhawat, Ajay Sharma, Harish Sharma, and Rajesh Kumar

Abstract Protein structure problem is a challenging task in the bioinformatics field; the most daunting task is to identify the accurate structure and bend angles in a stipulated time. As a known fact, it can be easily said that with the help of conventional methods, the solution to this problem cannot be found swiftly due to the complexity of the problem. In recent years, the development of metaheuristics and multipoint search techniques are proven an invaluable tool for solving this problem. In the light of this, the paper presents an application of a previously developed hybrid version of the Whale Optimization Algorithm (HWOASCA) to predicting the structure of proteins. An experiment bench of 8 sequences is formed and the optimization process is executed. Results reveal that the developed version yields better results in terms of statistical significance and independent run results.

Keywords Protein structure prediction problem (PSP) · Whale optimization algorithm · Sine cosine algorithm

A. Saxena · S. Shekhawat (✉)

Swami Keshvanand Institute of Technology, Management and Gramothan, Jaipur, India
e-mail: akash@skit.ac.in

A. Sharma

Government Engineering College, Jhalawar, India

H. Sharma

Rajasthan Technical University, Kota, India

e-mail: hsharma@rtu.ac.in

R. Kumar

Malaviya National Institute of Technology, Jaipur, India

e-mail: rkumar.ee@mnit.ac.in

1 Introduction

PSP is a complex and challenging optimization problem of Bioinformatics in which a stable protein structure with minimum energy has been found. Protein is an important physical part of any living organism and hence an attraction point of researches through many years [1]. HP lattice model [2] and AB OFF lattice model [3] have replaced classical methods like X-ray crystallography and Nuclear magnetic resonance spectroscopy which are not very useful due to heavy laboratory setup, financial and time limitations. AB off-lattice model is commonly known as Toy Protein Model in the field of bioinformatics. In recent years, the application of different metaheuristic algorithms has been reported to solve the PSP problem. Some of the pioneer approaches are based on Artificial Bee Colony and its variants [4, 5]. A fitness landscape-based analysis of different metaheuristic algorithms was conducted by [6]. Variants of Differential Evolution have been employed to solve PSP problems [7]. The nature of this problem is NP-hard. Authors in reference developed a hybrid algorithm by combining jDE and Hooke-Jeeves Direct Search for the prediction of the 3D AB OFF lattice model through Graphical Processing Units (GPUs). Authors conducted experiments on real protein sequences from Protein Data Bank (PDB) [8]. Further, chaotic variants of the Artificial Bee Colony Algorithm and Grasshopper Algorithm have been developed by Saxena et al. in works [9] and [10]. In addition to that, hybrid metaheuristics methods and their applications to PSP have been presented in reference [11]. A knowledge-based self-adaptive differential evolution algorithm has been employed for PSP in reference [12].

Recently, Jain et al. [13] presented an interesting experiment of combining trigonometric functions into the position update of the Whale Optimization Algorithm [14]. In a way, it is a fusion of the Sine Cosine Algorithm [15] into WOA. Authors have shown positive implications of this experiment through various mathematical analyses on real as well conventional problems. These applications motivated us to apply developed variant HWOASCA on the PSP problem. The following research objectives are framed for this study:

1. To develop an optimization routine and employ developed HWOASCA for predicting the structure of a protein by minimizing mean free energy values.
2. To evaluate the performance of developed HWOASCA and WOA on the bench of protein on the basis of statistical attributes (mean, standard deviation, maximum and minimum values).
3. To conduct various statistical significance tests and other relevant tests to judge the efficacy of the developed HWOASCA.

Remaining part of the manuscripts is organized as follows: in Sect. 2, problem formulation is presented. Section 3 describes the development of HWOASCA and Sect. 4 presents the results. Section 5 concludes the manuscript with future directions. In the following subsection, the problem formulation of PSP is explained.

2 Protein Structure Prediction Problem (PSP)

In this section, PSP is defined as an optimization problem where the accurate folding of the protein is found with the help of the accurate estimation of bend angles.

2.1 Problem Formulation

This subsection presents the Modelling of the AB-OFF Lattice model along with the mathematical details incorporated for the construction of the optimization problem. AB-OFF Lattice model has been used to describe the secondary structure of the proteins from years. This is based on the hypothesis that protein will fold in such a structure that possesses the lowest mean free energy value. For demonstrating the structure of the protein sequence, BABABBA is considered and folding of the sequence is shown in Fig. 1. As can be seen from the figure, the model incorporates 20 amino acids that can be classified as hydrophobic and hydrophilic residues. These residues are represented as ‘A’ and ‘B’. 2D realization of these amino acids can be seen from Fig. 1. These parts form a non-directional chain and the same are connected with chemical bonds. From the figure, it has also been observed that for minimization of energy value, accurate estimation of the bend angles are also inevitable. The structural arrangement of this chain is represented by $(n - 2)$ bend angles, i.e. $[\beta_2, \beta_3, \dots, \beta_7]$ and the optimization process can be aggregated in the direction of obtaining the same with the aim of minimizing the mean free energy value. From the above discussion, It is quite pragmatic to state that the range of these bend angles should be $[180^\circ, -180^\circ]$ that represents the rotation of sequence in clockwise and anti-clockwise rotation fashion. The free energy function of amino acid sequence can be given as per the following equation:

$$\text{Energy} = \sum_{i=2}^{n-1} \frac{(1 - \cos\beta_i)}{4} + 4 \sum_{i=1}^{n-2} \sum_{j=i+2}^n [p_{ij}^{-12} - C(\eta_i, \eta_j) p_{ij}^{-6}] \quad (1)$$

The property of the i_{th} individual particle is reflected by the values of $C(\eta_i, \eta_j)$; if residue is ‘A’ it will be equal to 1, otherwise it will attain -1 value. p_{ij} reflects the distance between i_{th} and j_{th} particles, and β_i is the bending angle.

$$p_{ij} = \sqrt{\left[1 + \sum_{k=i+1}^{j-1} \cos \left(\sum_{l=i+1}^k \beta_l \right) \right]^2 + \left[\sum_{k=i+1}^{j-1} \sin \left(\sum_{l=i+1}^k \beta_l \right) \right]^2} \quad (2)$$

$$C(\eta_i, \eta_j) = \frac{1}{8}(1 + \eta_i + \eta_j + 5\eta_i\eta_j) \quad (3)$$

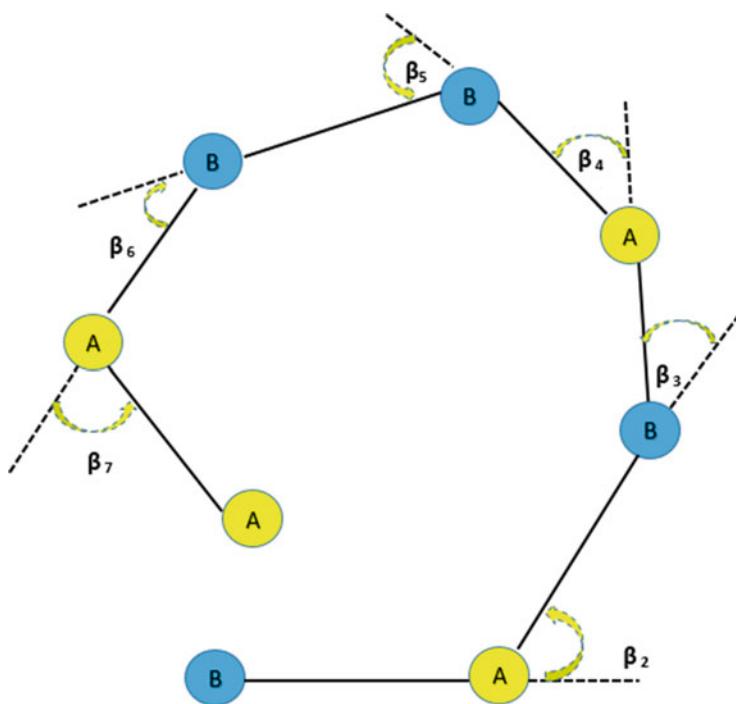


Fig. 1 2D representation of AB-OFF Lattice model for sequence 'BABABBBAA'

Table 1 Values of $C(\xi_i, \xi_j)$ for different pairs of protein

Pair configuration	$C(\xi_i, \xi_j)$
AA	1
BB	0.5
AB	-0.5
BA	-0.5

Equation 3 represents the interaction between two residues, i.e. A and B. Along with this, Table 1 exhibits the value of $C(\eta_i, \eta_j)$ for different pairs of residues.

Table 1 shows the correlation between different hydrophobic and hydrophilic residues. It is an established fact that the correlation between hydrophobic(A)-(A) particles are strongest and the value (1) is written in front of it. Likewise the weakest correlation is found between hydrophobic and hydrophilic residues. The values -0.5 are denoted for two different residues. After, following the steps described in this section, the protein folding problem can be transferred into a numerical optimization problem.

3 Hybrid Whale Optimization Algorithm

The Whale optimization Algorithm is now an established algorithm that was proposed by Mirjalili et al. [14]. WOA has been applied and tested on many engineering problems such as feature selection, strategic bidding problem of power sector [13]. WOA is based on the behaviour of Humpback whales and mathematical translations of the behaviour of whales. Sometimes, WOA is stuck in local minima and provides a poor convergence rate. To overcome this drawback, a hybridization of WOA with SCA is performed. This section presents the details of the Hybrid Whale Optimization algorithm. Jain et al. [13] presented a hybrid version of WOA and SCA for solving strategic bidding of the power sector. In the analysis, they revealed that by using trigonometric functions in the position update phase of WOA, substantial enhancement in convergence properties and optimization performance can be achieved. The whole algorithm can be illustrated in the following steps:

- Step 1: Initially, the whale chooses the existing solution as the best solution and it is near the prey. The other whales in the group update accordingly. This position updation can be mathematically given as

$$\vec{P} = \left| \vec{K} \vec{X}^* (t) - \vec{X} (t) \right| \quad (4)$$

$$\vec{X} (t+1) = \vec{X}^* (t) - \vec{A} \cdot \vec{K} \quad (5)$$

where X is the position vector and \vec{X}^* represents the best position vector, t is number of iterations, while \vec{A} and \vec{K} are constant vectors given by

$$\vec{A} = 2\vec{a}\vec{r} - \vec{a} \quad (6)$$

$$\vec{K} = 2\vec{r} \quad (7)$$

where \vec{r} is chosen between 0 and 1 randomly, and \vec{a} is linearly decreasing number from 2 to 0.

The search agents update their position by following the equation:

$$\vec{X} (t+1) = \begin{cases} R \times \sin(R) \times \vec{X}^* (t) - \vec{A} \cdot \vec{P}, & R < 0.5 \\ R \times \cos(R) \times \vec{X}^* (t) - \vec{A} \cdot \vec{P}, & R \geq 0.5 \end{cases} \quad (8)$$

Here, $R \in [0, 1]$ is a random number and acts as an identifying factor with \vec{A} .

- Step 2: In bubble net feeding, two different concepts have been used from WOA and SCA. The whale calculates its distance from the prey and forms a path spiral in shape which acts as a trap for its prey.

$$\vec{X}(t+1) = \vec{P}' \cdot e^{\alpha s} \cdot \cos(2\pi s) + \vec{X}^*(t) \quad (9)$$

$$\vec{P}' = \left| \vec{X}^*(t) - \vec{X}(t) \right| \quad (10)$$

- Step 3: When the prey did not get trapped in the spiral-shaped helix, then the search process becomes random. Before locating the prey, the whale used its best position but now it is updated randomly, which is actually exploration. Mathematically, we can write it as

$$\vec{X}(t+1) = \begin{cases} R \times \sin(R) \times \vec{X}_R - \vec{A} \cdot \vec{P}, & R < 0.5 \\ R \times \cos(R) \times \vec{X}_R - \vec{A} \cdot \vec{P}, & R \geq 0.5 \end{cases} \quad (11)$$

where $\vec{P} = \left| \vec{K} \vec{X}_R - \vec{X} \right|$. This solution is further used to find the best fit solution and in the optimization of our objective function.

4 Results

The results of the PSP problem are reported in Table 2. The statistical analysis in terms of mean, maximum (Max), minimum (Min) and standard deviation (SD) are shown. From the application of WOA and HWOASCA, the following conclusions can be drawn from this study:

1. It is observed that for small artificial protein sequences, the values of mean free energy are optimal for HWOASCA. The optimal values are shown in boldface. Values of SD and Max are also optimal for the first three small protein sequences. Hence, it can be concluded that HWOASCA outperforms WOA.
2. To make this analysis more meaningful, a statistical comparison based on the Wilcoxon rank-sum test results (p-values) is also shown in the last column of this table. We observed that the p-values are less than 0.05 that advocate a significant difference between HWOASCA and WOA.

4.1 Wilcoxon Rank-Sum Test

It is a known fact that firm conclusions cannot be derived on the basis of the mean of the independent runs and other statistical attributes of independent runs. Hence, for judging the optimization process, the Wilcoxon rank test is carried out for judging the significance over 5

Table 2 Results of different Protein Sequences

Sequence	Algorithm	Mean	Max	Min	SD	p-values
AAAAAB (5)	WOA	-1.46E+00	-1.23E+00	-1.59E+00	1.11E-01	2.43E-05
	HWOASCA [13]	-1.53E+00	-7.24E-01	-1.59E+00	1.68E-01	
AABAA (5)	WOA	-2.43E+00	-2.14E+00	-2.53E+00	1.08E-01	3.80E-03
	HWOASCA [13]	-2.49E+00	-2.33E+00	-2.53E+00	5.79E-02	
AABBAA (6)	WOA	-1.16E+00	1.27E-01	-1.94E+00	6.68E-01	5.00E-02
	HWOASCA [13]	-1.42E+00	1.27E-01	-1.97E+00	5.62E-01	
ABBABBABBBBAB (13)	WOA	-4.75E-01	1.13E-03	-1.32E+00	3.83E-01	1.20E-01
	HWOASCA [13]	-7.13E-01	-1.22E-02	-1.55E+00	3.99E-01	
BABBBAABBAAAAB (13)	WOA	-3.24E-01	-6.80E-02	-1.06E+00	3.16E-01	1.60E-03
	HWOASCA [13]	-6.26E-01	-9.15E-02	-1.51E+00	4.32E-01	
ABABBAABBBAAABBAABABAAB (21)	WOA	-1.05E+00	7.45E-02	-2.41E+00	7.64E-01	3.10E-01
	HWOASCA [13]	-1.37E+00	-4.54E-01	-2.77E+00	7.34E-01	

Further, it can be concluded that mean values are optimal in each case but p-values are especially less than 0.05 for sequences 1–3 and 5. This fact indicates that developed HWOASCA is capable of computing protein structures more efficiently.

4.2 Convergence Property Analysis

Convergence properties of HWOASCA and WOA are compared through convergence property analysis. This analysis is depicted through Fig. 2. We observe that the convergence for different sequences is accelerated with sine- and cosine-based mechanisms employed in the position update phase. The accelerated convergence properties can be seen in sequences 4, 5 and 6, and on the other hand, the function attains optimal values in the case of sequences 1, 2 and 3. It is to be noted here that these curves are the mean convergence curves obtained from independent runs.

4.3 Box Plot Analysis

For ensuring the optimization performance of the independent runs, often box plot analysis is conducted. Hence, for showcasing the efficacy of the HWOASCA, in optimizing the structure of proteins this analysis is conducted. Box plots for sequence 1–6 are added in Fig. 3. We observe that mean values are optimal in the case of HWOASCA and it is highlighted by an oval shape structure in the box plot. On the other hand, we found that Inter Quartile Ranges (IQRs) for these sequences are also very competitive in the case of HWOASCA. From this fact, we can easily

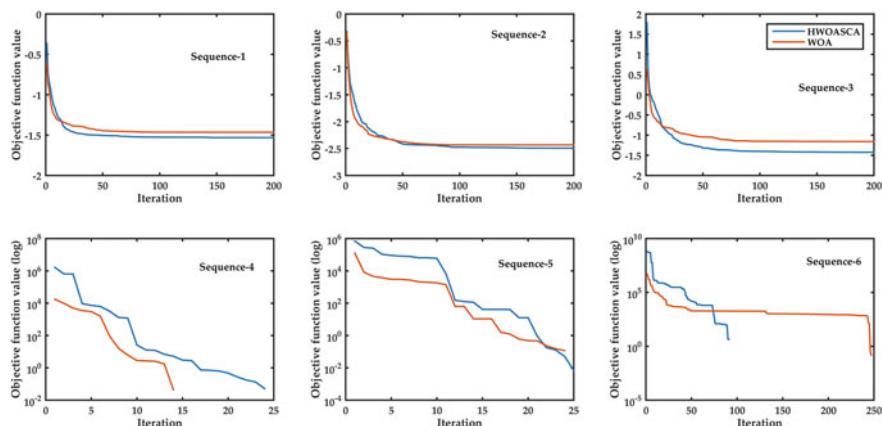


Fig. 2 Convergence property analysis

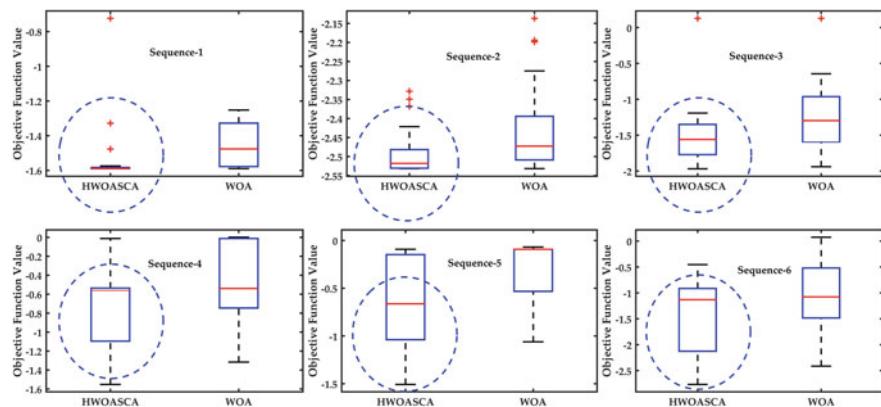


Fig. 3 Box plot analysis

say that along with conventional functions and power sector problems, developed HWOASCA is also very useful for optimizing the structure of proteins.

5 Conclusion

Identification of the correct structure of a protein is quite necessary and inevitable in the early detection of a disease and can be considered as a deciding factor while prescribing medication for critical diseases. In recent years, the application of nature-inspired algorithms are seen in this field prominently due to two reasons: first, they are adaptive and derivative-free and second, the response of these algorithms is faster as compared to conventional approaches. Keeping this fact in mind, the paper has presented an application of an already developed version of WOA and SCA algorithms to protein structure prediction. The following are the major conclusions:

1. On the basis of protein length, 6 different structures are chosen for optimizing the free mean energy values. We observed that the developed hybrid version can yield optimal values of mean free energy as compared to the parent algorithm.
2. For developing statistical significance, we have chosen the rank-sum test and we observe that optimization results of developed HWOASCA are quite distinct as compared to WOA.
3. Further, to showcase the efficacy of HWOASCA, property analysis and box plot analysis have also been conducted. From these analyses, it is quite evident that trigonometric function-based position update mechanism not only accelerates the convergence but also yields accurate results as seen from optimal quartile ranges of box plots.
4. Hence, it can be concluded that for solving PSP, developed HWOASCA can be a potential tool.

Acknowledgements The authors gratefully acknowledge the financial support from the CRS, RTU (ATU), TEQIP-III of Rajasthan Technical University, Kota, Rajasthan, India (Project Sanction No. TEQIP-III/RTU (ATU)/CRS/2019-20/33).

References

1. A. Venkatesan, J. Gopal, M. Candavelou, S. Gollapalli, K. Karthikeyan, Computational approach for protein structure prediction. *Healthc. Inform. Res.* **19**(2), 137–147 (2013)
2. K.A. Dill, S. Bromberg, K. Yue, H.S. Chan, K.M. Ftebig, D.P. Yee, P.D. Thomas, Principles of protein folding—a perspective from simple exact models. *Protein Sci.* **4**(4), 561–602 (1995)
3. F.H. Stillinger, T. Head-Gordon, C.L. Hirshfeld, Toy model for protein folding. *Phys. Rev. E* **48**(2), 1469 (1993)
4. B. Li, Y. Li, L. Gong, Protein secondary structure optimization using an improved artificial bee colony algorithm based on ab off-lattice model. *Eng. Appl. Artif. Intell.* **27**, 70–79 (2014)
5. B. Li, R. Chiong, M. Lin, A balance-evolution artificial bee colony algorithm for protein structure optimization based on a three-dimensional ab off-lattice model. *Comput. Biol. Chem.* **54**, 1–12 (2015)
6. N. Dulal Jana, J. Sil, S. Das, Selection of appropriate metaheuristic algorithms for protein structure prediction in ab off-lattice model: a perspective from fitness landscape analysis. *Inf. Sci.* **391**, 28–64 (2017)
7. B. Bošković, J. Brest, Protein folding optimization using differential evolution extended with local search and component reinitialization. *Inf. Sci.* **454**, 178–199 (2018)
8. M. Boiani, R.S. Parpinelli, A gpu-based hybrid jde algorithm applied to the 3d-ab protein structure prediction. *Swarm Evol. Comput.* **58**, 100711 (2020)
9. A. Saxena, R. Kumar, Chaotic variants of grasshopper optimization algorithm and their application to protein structure prediction, in *Applied Nature-Inspired Computing: Algorithms and Case Studies* (Springer, 2020), pp. 151–175
10. A. Saxena, S. Shekhawat, A. Sharma, H. Sharma, R. Kumar, Chaotic step length artificial bee colony algorithms for protein structure prediction. *J. Interdisc. Math.* **23**(2), 617–629 (2020)
11. N. Dulal Jana, S. Das, J. Sil, Protein structure prediction using improved variants of metaheuristic algorithms, in *A Metaheuristic Approach to Protein Structure Prediction* (Springer, 2018), pp. 169–195
12. P.H. Narloch, M. Dorn, A knowledge based self-adaptive differential evolution algorithm for protein structure prediction, in *International Conference on Computational Science* (Springer, 2019), pp. 87–100
13. P. Jain, A. Saxena, R. Kumar, Application and development of improved meta-heuristic for making profitable bidding strategy in a day-ahead energy market under step-wise bidding scenario. *Int. J. Swarm Intell.* **5**(2), 209–243 (2020)
14. S. Mirjalili, A. Lewis, The whale optimization algorithm. *Adv. Eng. Softw.* **95**, 51–67 (2016)
15. S. Mirjalili, Sca: a sine cosine algorithm for solving optimization problems. *Knowl.-Based Syst.* **96**, 120–133 (2016)

Clinical Named Entity Recognition Methods: An Overview



Naveen S. Pagad and N. Pradeep

Abstract Clinical named entity recognition plays an important role in the field of clinical research based on clinical information mining. The objective of clinical named entity recognition is to analyze and categorize medical conditions, namely symptoms, treatments, diseases, and body conditions in the Electronic Medical Records (EMRs). In recent years, deep neural networks have gained considerable achievement in several languages handling tasks and named entity recognition. Many algorithms are trained to learn the text features from the big scale labeled datasets. However, these data-driven techniques do not handle rare and unseen cases. Most of the existing methods have shown that human knowledge offered important information for managing rare and unobserved entities. However, there exist many issues in the medical records based on the clinical named entity recognition because of the various natural language text features and the exceptional clinical conditions in EMRs. Therefore, it is necessary to enhance the natural language text features of the model. Hence, this survey analyzes various methods of clinical named entity recognition. The main aim of this survey is to study the existing clinical named entity recognition techniques and classifies them under various categories. Accordingly, this paper gives a detailed survey of 25 research papers and classifies them under different categories, such as machine-learning-based methods, deep-learning-based methods, namely Neural Networks (NN), Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and active learning methods. Also, the analysis is promoted in the survey based on the publication year, research techniques, performance measures, and achievement of the research methodologies. Moreover, the problems in the methods are explained in the research gaps and issues. Furthermore, the future extent of this research work is provided based on the limitations identified from the existing research methods.

N. S. Pagad (✉)

Department of Information Science and Engineering, S.D.M Institute of Technology, Ujire, India

N. Pradeep

Department of Computer Science and Engineering, Bapuji Institute of Engineering and Technology, Davanagere, India

Keywords Clinical information mining · Clinical text · Named entity recognition · Natural language processing · Electronic medical records · Long Short-Term Memory · Deep learning

1 Introduction

With the growth of medical information development, Electronic Medical Record (EMR) has been extensively used in hospitals related to the medical research fields. EMR consists of the comprehensive and valuable medical-based information [1, 2] of the patients that are accessed and altered in digital formats [3]. The clinical text-based entities from the EMR are identified, and then the classification is done for the predefined categories, namely the symptoms and treatment of the diseases. In this record, the clinical texts create major impacts in healthcare research, namely disease inference, drug analysis, clinical decision support, and so on. The clinical texts can be investigated by the application of sequence labeling called the Named Entity Recognition [4]. Clinical named entity recognition is a fundamental process related to information extraction tasks and medical data mining operations [5]. Named entity recognition plays a vital role in identifying the medical named entities, namely the disease, body, disease symptoms, drugs, and disease treatment from the electronic medical records [6]. Clinical named entity recognition is the vital task in Natural Language Processing (NLP) for extracting the fundamental concepts called the named entities, such as the name of the disease, medication names, and the lab tests from the medical research records. Named entity recognition is an important NLP process in clinical research and translational research [3, 7, 8]. Named entity recognition sets a solid establishment in performing the tasks related to NLP and also named entity recognition finds the specific description, namely a geographical or the name of the individual.

For the analysis in the existing biomedical literature information, named entity recognition is considered as a major preprocessing phase to evaluate the drug entities, therapeutic entities, disease entities, genes, etc. Named entity recognition sets a solid foundation for other NLP tasks. Named entity recognition techniques are widely utilized in the medical field research areas [4], namely disease management, disease analysis, medical diagnosis, and prevention of disease [8]. The high-performance improvement in the named entity recognition systems is efficient in enhancing medical research [9]. The healthcare quality [10], clinical research enhancement, public health research works are improved by the objective assessment and automation based on the utilization of the high-performance tools concerning the capability based on forecasting and prediction. This can be efficiently achieved by the value extraction with the text-based massive data sets. Several approaches, such as machine learning [11], optimization algorithm [12] have been introduced for the sequence labeling task of named entity recognition. Among them, NNs have been gained popularity in training the named entity recognition as it does not depend on feature engineering and task-specific resources. NLP text-based mining methods and

machine learning approaches are some of the fundamental methods for constructing the named entity recognition models. However, some efficient strategies are required to handle the poor quality and excellence of the texts with the clinical jargon and abbreviations from the electronic medical records, and also these efficient strategies are utilized for the difficult interpretation of the discharge summaries [13]. Besides this, an Active Learning (AL) method tries to minimize the cost of annotation for the time by choosing valuable examples based on annotation. Moreover, the performance can be maximized by constructing statistical NLP models [14].

The main contribution of this research is to analyze various existing clinical named entity recognition mechanisms. The existing methods are categorized into machine-learning-based methods, deep-learning-based methods, and active-learning-based methods. The survey is about the deployed performance metrics for evaluation, publication year, and so on. Furthermore, the F-measure was reviewed for the performance evaluation of the clinical named entity recognition. Thus, it is considered as an inspiration for the future extension for devising effectual clinical named entity recognition techniques.

The organization of the paper is as follows: Sect. 2 reviews the existing clinical named entity recognition methods, Sect. 3 explains the research gaps and issues, Sect. 4 elaborates the analysis of the methods based on performance metrics, year of publication, and Sect. 5 concludes the paper.

2 Literature Review

This section deliberates the distinct methodologies adopted in the effective clinical named entity recognition. Accordingly, research papers are analyzed, and the clinical named entity recognition schemes practiced are widely categorized into three categories, machine-learning-based methods, deep-learning-based methods, and active-learning-based methods. Figure 1 depicts the categorization of the several approaches employed in the clinical named entity recognition. The existing research works related to the categorized clinical named entity recognition are discussed below as follows:

2.1 Machine-Learning-Based Methods

This subsection elucidates the machine-learning-based methods adopted in the different research works as follows. Ghiasvand and Kate [15] introduced a machine learning approach for the clinical named entity recognition, and this approach does not utilize any type of manual annotations. This method utilized raw corpus and the record of semantic types with the resources like a Unified Medical Language System (UMLS) and so on. In this method, annotations are generated by considering these resources for training the machine learning approaches to determine their boundary

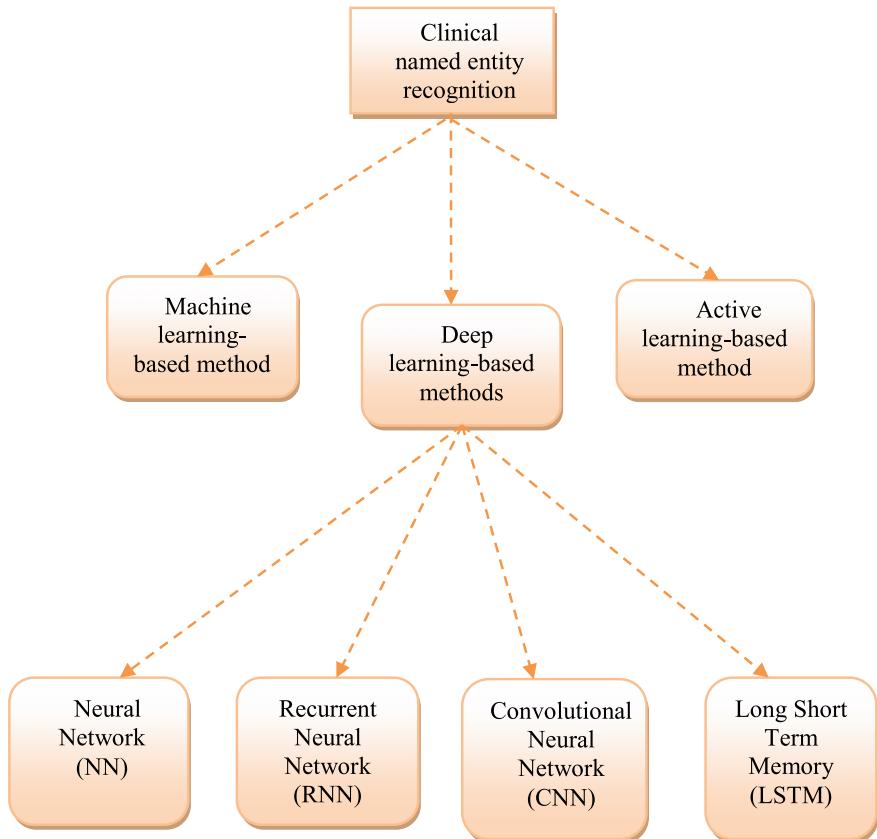


Fig. 1 Categorization of clinical named entity recognition methods

and also to evaluate the named entity. This method does not use the costly manual annotation, thereby provided better performance in constructing the named entity recognition-based systems for the clinical-based domains.

Boag et al. [16] developed clinical named entity recognition named CliNER to enhance the performance of the named entities. This method was considered as the two-pass supervised system based on machine learning. The first pass utilized the Conditional Random Fields (CRF)-based linear chain mechanism so that the concepts for the boundaries were identified. Using the results obtained from the first pass, phrases based on the clinical model were determined, and also several text-based features were utilized in this method. The concepts in the text can be recognized using the annotated training data, thereby achieving better-optimized performance results. Tang et al. [17] developed Structural Support Vector Machines (SSVMs) using rich features to recognize the named entities in the hospital discharge summaries. The representation based on the word features were determined for the

clustering approach in the SSVMs recognizer, and then these evaluated rich features were combined, thereby resulting in better results.

Tang et al. [18] designed an SSVM using the clinical named entity recognition method to detect the medical entities for the hospital discharge summaries. In this method, two kinds of word representations were extracted, namely the feature based on distributional representation, and the clustering-based representation features. These features were then combined with the SSVM-based clinical named entity recognition systems. The representation of the words concerning their features was evaluated to generate better output performance. Mao et al. [19] introduced the Hidden Markov Model (HMM) for recognizing the clinical named entities. In this model, the training corpus involved the manual annotations, and this trained corpus was utilized to train the HMM model for performance improvement. This model was employed for segmenting the word-based text corpus in such a way that the named entities with dissimilar suffixes were generated and finally rate-based on precision and recall values were calculated and determined.

2.2 Deep-Learning-Based Methods

In this subsection, the deep learning approaches practiced for clinical named entity recognition are portrayed below.

2.2.1 Neural Networks

This subsection elucidates the NN-based methods adopted in the different research works as follows, Xu et al. [4] developed an attention-based NN model to control the global information related to the document level in such a way the problems can be reduced. The global information was generated from the document-based representations related to the pretrained Bi-directional Language Model (Bi-LM) and the neural attention. The unlabeled data was utilized by the parameters from the pretrained Bi-LM, and then the data was transferred to the named entity recognition for enhancing the performance results.

Zhang et al. [6] developed an overlapping NN for the clinical named entity recognition and this method contains three modules, namely embedding module, overlapping module, and the CRF module. In the embedding module, the input data was transformed into vector-based sequences through querying the characters in the scale. In the overlapping module, the text sequence based on the Long Short-Term Memory (LSTM) and the Gated Recurrent Unit (GRU) was encoded, and finally, in the CRF module, the series of optimal tags were obtained by decoding the module output to get efficient results.

Gridach [20] devised a character-level neural network for the recognition of named entities. This neural network was designed and performed based on the Bi-LSTM, character-level embeddings, embeddings based on the pretrained word and CRF. This

type of strong deep NN was utilized, and hence capable of constructing a method with no use of gazetteers and dictionary so that the tasks based on feature engineering were eliminated. The pretrained word embeddings were integrated with the embeddings based on the character-level for capturing the orthographic information and the morphological information effectively, which may decrease the issues based on the Out-Of-Vocabulary (OOV).

Zhao et al. [21] introduced a framework, called the deep neural multi-task learning framework in cooperation with the medical entities based normalization and recognition. In this framework, the common representations of both the recognition and the normalization tasks were done using multi-task learning. The feedback strategies from the low-level to the high-level tasks were incorporated in such a way that the hierarchical tasks were successfully converted into the parallel multi-task setting so that mutual supports were maintained among the multi-tasks.

Liu et al. [22] developed a Deep Neural Network (DNN) approach, which integrates the Bi-directional Long Short-Term Memory (Bi-LSTM) and the CRF to accurately recognize and extract the named entities. Initially, the information from the EMR was defined to evaluate and extract the named entities, and then a DNN-based Continuous Bag Of Word Clusters (CBOWC) was utilized for transforming the vectors based on the single-dimensional word into the spatial word vectors with the distributed model. Finally, the Bi-LSTM-CRF approach concerning the online clinical data was employed in such a way that the vectors with the text features were converted by the spatial word vectors so that the complete knowledge about the recognition of online medical data analysis was realized.

Luu et al. [13] introduced a framework called multilevel named entity recognition framework for the automatic recognition of the clinical entities. This framework was designed by considering different levels of information, and at every level, the issues based on the higher complexity in the named entity recognition tasks were handled and resolved. Level 1 named entity recognition layer handled the named entity recognition tasks with lower complexity, where fewer entities with several labeled data were automatically recognized to build the named entity recognition models. Level 2 named entity recognition layer handled the difficult tasks, where the sparse data with the limited labeled training data were recognized to build the named entity recognition models, thereby improving the performance of the framework.

2.2.2 Recurrent Neural Networks

The different researches adopting the RNN-based techniques for the clinical named entity recognition are elucidated in this subsection. Lerner et al. [23] introduced a terminology-based approach to enhance the recognition results of the supervised model for the clinical-based terminologies. Initially, an annotated corpus was constructed to determine the named entities and then, three various approaches were employed. The terminology-based approach was constructed for the clinical management based on the Unstructured Information, and then the supervised-learning method was utilized with the bi-Gated Recurrent Unit-CRF (GRU-CRF). Finally,

the hybrid system was utilized to detect the named entities, such as the name of the drug, disease symptoms and disorders, policies based on diagnosis, and therapeutic rules.

Luu et al. [24] developed a method for the clinical named entity recognition using deep machine learning approaches. The deep learning approaches, namely RNN and the Feed-Forward Networks (FFN) were utilized for improving the recognition results of the named entities. In the preprocessing stage, various features based on the NLP were considered, and then in the feature extraction stage, the needed features were extracted using the word2vec model. The poor excellence in the data and the complex clinical tasks can be improved by these methods, thereby resulting in improved results.

2.2.3 Convolutional Neural Networks

This subsection elaborates on the different researches employed with the CNN-based methods. Wu et al. [8] designed a deep learning approach, called the CNN, and RNN for extracting the clinical text-based concepts in such a way, the deep learning approaches were utilized with the three CRFs baselines. In this approach, the RNN model was trained by considering the word-based embeddings concerning the features based on unsupervised learning and the manually defined features. This method achieved better performance for the medical-based extraction of concepts, which involves learning based on automatic features, a representation based on the distributed feature, whereas the output-based on long-term dependency.

2.2.4 Long Short-Term Memory

In this subsection, the various researches adopting the LSTM-based techniques for the clinical named entity recognition are elucidated.

Khan et al. [25] developed a hybrid LSTM-CRF method for generating the name of the disease. This hybrid LSTM-CRF consists of two layers, namely the feature representation layer and the bi-directional LSTM-CRF-based layer. Initially, the layer of feature representation was utilized for representing a sentence in such a way that the embeddings based on context, word, domain knowledge, and character were concatenated for extracting the features. These extracted features were subjected to the bi-directional LSTM-CRF layer to tag the sentences based on the clinical texts.

Cho et al. [26] introduced a combinatorial feature embedding using the Bi-directional Long Short-Term Memory (Bi-LSTM) with CRF. In this method, the representations based on the characters were extracted from the Bi-LSTM and CNN and so that these extracted representations were combined in such a way that the performance of the method was enhanced. Moreover, the attention-based mechanism was utilized with this method to reduce the issues based on the long-term dependency of the LSTM, thereby resulting in better recognition of the entities.

Xu et al. [27] developed a Bi-directional LSTM method based on Conditional Random Field (Bi-LSTM-CRF) for medical named entity recognition. This architecture consists of three layers, namely the Bi-LSTM layer with character, the Bi-LSTM layer with words, and the CRF layer. The representations based on the character were utilized for improving the clinical named entity recognition for the predefined CRF layer. In the first layer, the word expressions-based character level in the text was learned. In the second layer, the word-based embeddings were generated by integrating the networks of word-based lookup table, with the multiple word-based Bi-LSTM. The third layer captured the extracted label relationships from the CRF model.

Xu et al. [9] developed Document-level Attention-based Bi-LSTM-CRF (DABLC) for recognizing the disease entities. In this method, the entities were matched with the disease dictionary by utilizing the mechanism based on string matching where the dictionary was generated concerning the disease ontology. Besides this, the dictionary-based attention layer was constructed using DABLC employing integrating the mechanism based on the document-level attention and the matching strategy based on the disease dictionary. Finally, the results for the medical named entity recognition were generated by the integration of the disease dictionary and the proposed Bi-LSTM-CRF.

2.3 Active-Learning-Based Methods

This subsection elaborates on the different researches employed with the active-learning-based methods. Wei et al. [28] developed a novel algorithm called the cost-aware active learning algorithm (Cost-CAUSE) to annotate the clinical named entities. Initially, the syntactic and the lexical features were used for evaluating the cost based on the annotations, and then the measure of the cost was integrated with the active learning approaches so that this algorithm saved the cost based on the annotation for the random sampling. Chen et al. [14] designed an active-learning-enabled annotation system to construct the clinical named entity recognition models. This method utilized the annotated sentences, and then the next sentence for the annotation was selected iteratively for recognizing the clinical named texts. Various querying algorithms were implemented and examined to obtain better recognition results. This active learning system enables a gap-free experience in annotating the sentences.

2.4 Other Clinical Named Entity Recognition Methods

In this section, the different researches adopting the other techniques for the clinical named entity recognition are elucidated. Zhang and Elhadad [29] devised an unsupervised framework for generating the clinical-based entities from the clinical-based texts. This framework consists of the extractor-based seed term, NP chunker, Inverse

Document Frequency (IDF)-based filter, and distributional semantics-based classifier. In this method, lexical-based semantics, and the shallow syntactic-based analysis was employed in such a way that this technique does not depend on heuristics rules, policies, and any training data, so that this framework can be used in various applications. Once the candidates were filtered concerning the IDF-based filter, the classifier in this framework exhibits significant improvement in the performance based on the classification of the entities. Urbain [30] introduced a multi-stage NLP system for the named entity recognition by utilizing rule-based logic and Bayesian statistics to determine the heart disease based on the risk factor events. This method achieved better performance in terms of accuracy for identifying the named entities. However, this method may suffer from inaccuracies when specifying the event attributes. These inaccuracies can be handled by utilizing the distributional semantic model to specify the risk factors based on heart disease. Keretna et al. [31] developed a classification-based named entity recognition method, named extended Segment Representation (SR) for improving the recognition of named entity in medical-related applications. In this method, words are allocated with a new class that appears as a named entity in one context and does not belong to the named entity in other contexts. In these cases, ambiguity affected the output obtained from the classification-based named entity recognition system so that an individual new class was assigned for every word in such a way that the named entity recognition ambiguity permits the classifier to recognize the named entities accurately and more effectively, thereby maximizing the classification accuracy.

Zhang et al. [5] introduced a method, named Category-Multi-Representation (CMR) to capture the semantic relationship among the clinical category and words concerning different views. This method utilized the huge scale unannotated corpus and the annotated data with small sets, which may reduce the issues caused by human attempts. Initially, semantic clinical space was constructed by training the method of word embedding. The categories, which were predefined may be considered as a cluster of words in the space and the abstract representations were derived from various perspectives for each category. Then, the new features were generated for the target word concerning the CMR distance. Finally, the learning algorithm was applied for the newly generated features for estimating the method. Wang et al. [32] developed a framework called the Label-aware Double Transfer Learning (La-DTL) for the cross-specialty recognition of named entity in such a way that the designed framework for one particular specialty was presented to another specialty along with some efforts based on limited annotation. In this framework, every sentence with the input was transformed into series of embedding vectors and these embedded vectors were subjected to the Bi-LSTM so that the contextual information was encoded successively into the hidden vectors with the fixed length. Also, the Bi-LSTM layers and the embedded vectors were shared among the domains, namely source and target. The hidden vectors were presented into the particular CRF layers with a source and target domain so that the sequence of labels is predicted. Domain-constrained CRF layers were utilized for improving the performance of the target domain.

3 Research Gap and Issues

This section elaborates on the research gaps and issues of various categorized methods. The challenges faced by the machine-learning-based methods are given below: In [15], it is necessary to remove the constraints in the noun phrases for enhancing the results of machine learning approaches. In [16], the challenge lies in utilizing the CliNER method in a large community to make the performance improvements by narrowing the break among the clinical domain and the general-domain named entity recognition. In [17], the challenge lies in integrating the results of SSVMs and CRFs for the clinical named entity recognition systems for effective clinical text processing. In [18], the devised method failed to utilize the word representation features to improve the named entity recognition performance. The devised HMM does not enlarge the test set and training set to enhance the scope and efficiency of the model [19].

The research issues in the deep-learning-based methods were as follows, the developed attention-based neural network architecture failed to consider the unlabeled data with the Bi-LM, which was pretrained to efficiently improve the performance [4]. The challenge lies in investigating the deep learning methods and applying these methods in clinical NLP systems for the better extraction of clinical texts [8]. In [23], the devised terminology-based system method does not consider the large pretrained language approaches to offer an efficient contextualized word representation by solving the issues related to the low annotation regime. The devised document-level attention-based Bi-LSTM-CRF failed to consider the deep-learning-based NER methods to progress the poor-quality clinical text notes [13]. In [24], it is necessary to investigate deep learning configurations, namely CNN and LSTM concerning the feature selection techniques to enhance the automatic clinical name entity recognition performance, and also to offer support for the clinical contexts in real-world applications. In [25], the challenge lies in obtaining the deeply contextualized embeddings of the clinical trial texts by utilizing the deep bi-directional transformer-based language methods. The developed combinatorial feature embedding using CNN and LSTM approach does not use the knowledge transferring approach for enhancing the performance of bio-NER and to resolve the issues related to error analysis [26]. In [9], it is necessary to employ deep learning models and a special disease named entity recognition to address the long sentences.

Following are the research issues in the active-learning-based methods: In [28], the cost-aware active learning method does not examine the indirect and direct factors to improve the active learning approaches for effective results. In [14], it is necessary to employ active learning algorithms for the precise evaluation of the annotation time and large-scale problems. The research issues in the other clinical named entity recognition methods are as follows. In [29], the challenge lies in removing the errors using the parsing technique and then selecting the noun phrases as the candidates in the parse tree. In [30], it is necessary to use the larger electronic health record dataset for collecting distributional statistics information to obtain a clear evaluation of the method. The developed SR method failed to integrate the various named entity

Table 1 Analysis concerning the published year

Publication year	Number of research papers
2020	3
2019	5
2018	5
2017	6
2015	3
2013	2
2012	1

recognition techniques for improving the results [31]. In [5], the challenge lies in estimating the developed category multi-representation method in biomedical fields, and also this method failed to employ new unsupervised methods to enhance the performance of the training data sets. The devised Label-aware Double Transfer Learning (La-DTL) failed to consider the extraction of structural information related to the cross-specialty media by recognizing and linking the named entities for the improved output performance [32].

4 Analysis and Discussion

The analysis and discussion of clinical named entity recognition using various research papers based on the categorization of methods, publication year, and performance evaluation metrics are elaborated in this section.

4.1 Analysis Using Published year

The review for the clinical named entity recognition using the publication year of several 25 research papers is illustrated in this section. The analysis concerning the published year is depicted in Table 1. Out of the 25 papers surveyed, more number of research papers were published in 2017.

4.2 Analysis Using Methods

This section illustrates the review concerning various clinical named entity recognition methods. The various methods used for the clinical named entity recognition are depicted in Fig. 2. Based on Fig. 2, it is noted that 25% of the research papers used machine learning approaches, the deep learning approaches were used in 65%

Analysis based on techniques

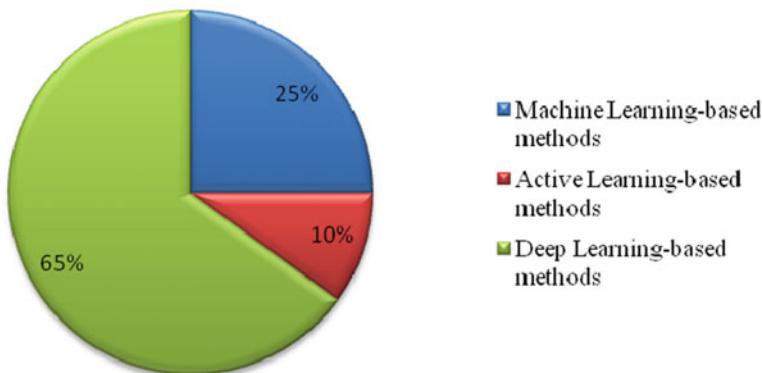


Fig. 2 Analysis using the classification of clinical named entity recognition methods

of the researches, and 10% of the researches utilized active learning-based methods. Therefore, from the analysis, deep-learning-based methods are widely developed techniques for clinical named entity recognition.

4.3 Analysis Using Evaluation Metrics

The evaluation metrics are analyzed and discussed in this section. Precision, Recall, F-measure, and accuracy are the evaluation metrics considered for clinical named entity recognition. From Table 2, it is clearly shown that F-measure is the most commonly used performance metric.

Table 2 Analysis in terms of evaluation metrics

Performance metrics	Number of Research papers
Precision	[4, 5, 6, 8, 9, 13, 15, 16, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27, 29, 30, 31]
Recall	[4, 5, 6, 8, 9, 13, 15, 16, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27, 29, 30, 31]
F-measure	[4, 5, 6, 8, 9, 13, 14, 15, 16, 17, 18, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32]
Accuracy	[25]

Table 3 Analysis in terms of F-measure

Range	F-measure
50–60%	[29, 32]
60–70%	[13, 14, 15, 24]
70–80%	[16, 27, 28]
80–90%	[4, 5, 8, 9, 17, 18, 20, 21, 22, 25, 26, 30]
90–99%	[6, 23, 31]

4.4 Analysis Using Values of Performance Metrics

The analysis using performance metrics value is illustrated in this section. The analysis using F-measure is explained in this section.

4.4.1 Evaluation Based on F-measure

In this section, the evaluation based on F-measure is explained. Table 3 shows the review based on F-measure scores is specified by five ranges as, 50–60%, 60–70%, 70–80%, 80–90%, and 90–99%. From the below table, it is shown that the research article [6, 23, 31] achieved enhanced F-measure and [29, 32] research papers had low F-measure value.

5 Conclusion

This paper provides a survey of the various clinical named entity recognition methods. Initially, 25 research works in the field of clinical named entity recognition are gathered from the Google Scholar, ScienceDirect and IEEE Xplore. These research works were categorized based on the approaches, namely machine learning, deep learning techniques, such as NN, RNN, CNN, LSTM, and active learning. The research articles are reviewed, analyzed, and the research gaps and the issues practiced by the recent research papers are explained. Furthermore, the analysis and discussion of the survey are explained based on the classification methods, evaluation metrics, and publication year. From the analysis, it is noted that the deep learning approaches are the commonly utilized method in most of the research papers, and also the evaluation metric F-measure is the most utilized metric in many of the research papers. Moreover, the survey suggested the future scope for the clinical named entity recognition by considering various research gaps and the problems of the existing methods.

References

1. J. S. Raj, J. Shobana, I. V. Pustokhina, D. A. Pustokhin, D. Gupta, and K. Shankar, “Optimal feature selection-based medical image classification using deep learning model in internet of medical things”. *IEEE Access*. **8**, 58006–58017 (2020)
2. I. V. Pustokhina, D. A. Pustokhin, D. Gupta, A. Khanna, K. Shankar, and G. N. Nguyen, “An effective training scheme for deep neural network in edge computing enabled internet of medical things (iomt) systems”, *IEEE Access*. **8**, 107112–107123 (2020)
3. S. M. Meystre, G. K. Savova, K. C. Kipper-Schuler, J. F. Hurdle, “Extracting information from textual documents in the electronic health record: a review of recent research”, *Yearb. Med. Inform.* 128–144 (2008)
4. G. Xu, C. Wang, X. He, “Improving clinical named entity recognition with global neural attention”, in *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, (2018), pp. 264–279
5. J. Zhang, J. Li, S. Wang, Y. Zhang, Y. Cao, L. Hou, X. L. Li, “Category multi-representation: a unified solution for named entity recognition in clinical texts”, in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, (2018) pp. 275–287
6. R. Zhang, Y. Gao, R. Yu, R. Wang, W. Lu, Medical named entity recognition based on overlapping neural networks. *Procedia Computer Science* **174**, 27–31 (January 2020)
7. P.M. Nadkarni, L. Ohno-Machado, W.W. Chapman, Natural language processing: an introduction. *J. Am. Med. Inform. Assoc.: JAMIA* **18**(5), 544–551 (2011)
8. Y. Wu, M. Jiang, J. Xu, D. Zhi, H. Xu, Clinical named entity recognition using deep learning models. *AMIA Annu. Symp. Proc., Am. Med. Inform. Assoc.* **2017**, 1812 (2017)
9. K. Xu, Z. Yang, P. Kang, Q. Wang, W. Liu, Document-level attention-based BiLSTM-CRF incorporating disease dictionary for disease named entity recognition. *Comput. Biol. Med.* **108**, 122–132 (May 2019)
10. S. Sheeba Rani, J. A. Alzubi, S. K. Lakshmanaprabu, D. Gupta, R. Manikandan, “Optimal users based secure data transmission on the internet of healthcare things (IoHT) with lightweight block ciphers” *Multimedia Tools Appl.* **79**, 35405–35424 (2020)
11. J. Alzubi, Optimal classifier ensemble design based on cooperative game theory. *Res. J. Appl. Sci. Eng. Technol.* **11**(12), 1336–1343 (2015)
12. J. Sethuraman, J. A. Alzubi, R. Manikandan, M. Gheisari, and A. Kumar, “Eccentric methodology with optimization to unearth hidden facts of search engine result pages”. *Recent Patents On Computer Science*, **12**(2) (2019)
13. T. M. Luu, R. Phan, R. Davey, G. Chetty, “A multilevel NER framework for automatic clinical named entity recognition”, in *Proceedings of IEEE International Conference on Data Mining Workshops (ICDMW)*, (2017) pp.1134–1143
14. Y. Chen, T.A. Lask, Q. Mei, Q. Chen, S. Moon, J. Wang, K. Nguyen, T. Dawodu, T. Cohen, J.C. Denny, H. Xu, An active learning-enabled annotation system for clinical named entity recognition. *BMC Med. Inform. Decis. Mak.* **17**(2), 35–44 (July 2017)
15. O. Ghiasvand, R.J. Kate, Learning for clinical named entity recognition without manual annotations. *Informatics in Medicine Unlocked* **13**, 122–127 (January 2018)
16. W. Boag, K. Wacome, T. Naumann, A. Rumshisky, “CliNER: a lightweight tool for clinical named entity recognition”, *AMIA joint summits on clinical research informatics*, (2015)
17. B. Tang, H. Cao, Y. Wu, M. Jiang, H. Xu, “Clinical entity recognition using structural support vector machines with rich features”, in *Proceedings of the ACM Sixth International Workshop on Data and Text Mining in Biomedical Informatics*, (2012) pp.13–20
18. B. Tang, H. Cao, Y. Wu, M. Jiang, H. Xu, Recognizing clinical entities in hospital discharge summaries using structural support vector machines with word representation features. *BMC Med. Inform. Decis. Mak.* **13**(S1), S1 (April 2013)
19. X. Mao, F. Li, H. Wang, H. Wang, “Named entity recognition of electronic medical record based on improved HMM algorithm”, in *Proceedings of International Conference on Computer Technology, Electronics and Communication (ICCTEC)*, (2017) pp. 435–438

20. M. Gridach, Character-level neural network for biomedical named entity recognition. *J. Biomed. Inform.* **70**, 85–91 (June 2017)
21. S. Zhao, T. Liu, S. Zhao, F. Wang, A neural multi-task learning framework to jointly model medical named entity recognition and normalization. *Proc. AAAI Conf. Artif. Intell.* **33**, 817–824 (July 2019)
22. X. Liu, Y. Zhou, Z. Wang, Recognition and extraction of named entities in online medical diagnosis data based on a deep neural network. *J. Vis. Commun. Image Represent.* **60**, 1–5 (April 2019)
23. I. Lerner, N. Paris, X. Tannier, “Terminologies augmented recurrent neural network model for clinical named entity recognition”. *J. Biomed. Inform.* **102**, 103356 (2020)
24. T. M. Luu, R. Phan, R. Davey, G. Chetty, “Clinical named entity recognition based on recurrent neural networks”, in Proceedings of 18th International Conference on Computational Science and Applications (ICCSA), (2018) pp.1–9
25. M. A. Khan, M. Shamsuzzaman, S. A. Hasan, M. S. Sorower, J. Liu, V. Datla, M. Milosevic, G. Mankovich, R. van Ommering, N. Dimitrova, “Improving disease named entity recognition for clinical trial matching”, in Proceedings of IEEE International Conference on Bioinformatics and Biomedicine (BIBM), (2019) pp.2541–2548
26. M. Cho, J. Ha, C. Park, S. Park, “Combinatorial feature embedding based on CNN and LSTM for biomedical named entity recognition”, *J. Biomed. Inform.* **103**, 103381, (2020)
27. K. Xu, Z. Zhou, T. Hao, W. Liu, “A bidirectional LSTM and conditional random fields approach to medical named entity recognition”, in International Conference on Advanced Intelligent Systems and Informatics, (2017) pp.355–365
28. Q. Wei, Y. Chen, M. Salimi, J.C. Denny, Q. Mei, T.A. Lasko, Q. Chen, S. Wu, A. Franklin, T. Cohen, H. Xu, Cost-aware active learning for named entity recognition in clinical text. *J. Am. Med. Inform. Assoc.* **26**(11), 1314–1322 (November 2019)
29. S. Zhang, N. Elhadad, Unsupervised biomedical named entity recognition: experiments with clinical and biological texts. *J. Biomed. Inform.* **46**(6), 1088–1098 (December 2013)
30. J. Urbain, Mining heart disease risk factors in clinical text with named entity recognition and distributional semantic models. *J. Biomed. Inform.* **58**, S143–S149 (December 2015)
31. S. Keretna, C.P. Lim, D. Creighton, K.B. Shaban, Enhancing medical named entity recognition with an extended segment representation technique. *Comput. Methods Programs Biomed.* **119**(2), 88–100 (April 2015)
32. Z. Wang, Y. Qu, L. Chen, J. Shen, W. Zhang, S. Zhang, Y. Gao, G. Gu, K. Chen, Y. Yu, “Label-aware double transfer learning for cross-specialty medical named entity recognition”, pp.1804–09021, (2018)

Mobile Phone SMS Notification Behavior Analysis Using Machine Learning Technique



Sumaiya Deen Muhammad, Farzana Tasnim, and Sanjida Sharmin

Abstract At present, mobile phone is considered to be a device through which users are always reachable to be communicated wherever and whenever needed via phone calls, SMS, or messaging applications, for instance, WhatsApp, Messenger, etc. Sometimes this accessibility makes people annoyed particularly when someone receives unnecessary and inappropriate notifications in inopportune time (e.g., meeting, seminar, workshop, etc.). Almost every day, we receive bundles of unnecessary SMS from mobile operators and different brands promoting various offers. It distracts us while we are busy with important tasks. Sometimes we miss important messages due to these kinds of promotional messages. This paper presents a prediction model to classify SMS notifications based on users' preferences. Comparing different machine learning techniques, we have found random forest algorithm gives the highest accuracy (85%).

Keywords Context-awareness · Smartphone intelligence · Mobile data science · Mobile operator SMS · Mobile notification behavior · Machine learning technique · SMS marketing

S. Deen Muhammad (✉) · F. Tasnim · S. Sharmin

Department of Computer Science and Engineering, International Islamic University Chittagong, Chattogram 4318, Bangladesh

e-mail: sumaiya.muhammad@iiuc.ac.bd

F. Tasnim

e-mail: tasnim@iiuc.ac.bd

S. Sharmin

e-mail: ssharmin@iiuc.ac.bd

1 Introduction and Background

Due to the enormous demand for smartphones and their massive usage throughout daily activities, various context-aware technologies have been adopted in mobile devices. People use mobile phones, particularly smartphones as the primary mode of communication because of their portability facility, reduced size and easy-to-use features, and an incredible amount of smart mobile applications. Usage of smartphones has now surpassed the use of feature phones [1]. According to data-portal.com, a recent survey shows that 5.15 billion people are using unique mobile phones in the world where 59% of the total population is using the internet [2]. Hence, research and experimentation on context-aware mobile phone data are being carried out in a broad range.

Nowadays, smartphone users receive various kinds of information, for instance, notifications from different mobile applications, reminders from scheduled tasks, SMS (short message service) from different personnel, operator SMS, promotional SMS from different organizations, brands, etc. Often these notifications arrive at inopportune moments while users are at office, meeting with boss, or attending seminar. Commonly people ignore SMS at those moments. This paper investigates users' predilection according to situations where contextual data like social situation, day, time, social relationship has been analyzed implementing machine learning techniques.

2 Literature Review

Many researchers already contributed a lot to this area. We discuss some remarkable research works which relate to our paper.

Schilit et al. [3] defined context-aware computing applications as systems that acclimate in consonance with a device's location, social situation, its hosts, other accessible devices, and adjacent people. They [3] identified three features of contexts: who we are, who we are with, and our nearby resources which indicate that context not only comprises the device's location but also its surroundings are important since other factors around us are moving. According to Kang et al. [4], a context-aware system is to learn ubiquitous computing in a static set of states, for instance, workplace, home, etc.

For the last couple of decades, many research works have been done on context-aware recommendations for portable smart devices such as laptops, tablets, smartphones, smartwatches, etc. For example, Cheverst et al. [5] developed and evaluated an intelligent electronic tourist guide, which provides context-aware information to the visitors. Mobile computing technologies have been used along with wireless infrastructure in this system. In [6], a novel approach has been proposed to conduct mining personal context-aware preferences according to context data of

the handlers, and using the data based on users' preferences, personalized context-aware preferences have been represented to the users. Chen and Kotz [7] investigated certain context-aware applications, for example, call forwarding, teleporting, shopping assistant, cyber guide, etc., where device's location, current time, etc., contexts have been used thoroughly. [8] introduced a mining engine that mines phone usage sequential patterns automatically using mobile user's contextual data. This engine provides device intelligence by implementing mined longitudinal patterns. Mehrotra et al. [9] presented an interruptibility management application to manage mobile apps notifications where mining association rules have been applied.

In order to obtain context data from smartphones to classify their behavior, a number of researchers analyze certain periods of log information, for instance, phone call log, battery lifelog, location history, web search log, mobile applications log, SMS log, mobile applications notification log, app usage period log, etc. [10]. Mehrotra et al. [9] worked on an in-the-wild dataset that consists of 11,185 notifications from 18 users to develop an intelligent system to recognize useful notifications and to decline if a message will be declined by the user. 3,174 notification data have been used in a study on notification management [11] where notification arrival date, time, notification title, notification message, and application package name has been plotted as features. As stated in [12], contextual data like location, phone events, notification interaction events, etc., have been used to design a notification manager. Boase and Ling [13] have used server log data consists of outgoing call and SMS events. An in situ study has been conducted by analyzing real-world smartphone notifications log data, collected from 15 mobile users [14]. 120 million phone use events data has been analyzed [15] to predict the right moment to deliver a notification to a user whether the user will check it or ignore it. A dataset consists of 55,105 phone call records has been investigated in [10] to build a personalized context-based behavior prediction model.

Several authors conducted research on push notifications, mobile application notifications, etc., where context data played a vital role to build a prediction model. To design mobile intelligent systems, often various machine learning algorithms are employed. Peilot et al. [15] have applied a gradient boosting regression tree algorithm to form a prediction model that predicts the best-fitted time to make users occupied with notifications and their content. Constraint-based matrix factorization algorithm has been carried out in [6] to predict context-aware preferences for the users. As indicated in [10], recency-based rules have been exerted for the prediction model. Sarker et al. [16] proposed an upgraded Naive Bayes classifier method to detect noise to classify users' phone call behavior accurately.

3 Methodology

3.1 Overview of Dataset

We have conducted a survey regarding users' preferences of SMS notification and ringtone based on their social situations and social relationships.

The dataset includes professional individuals who often remain busy with meeting, office work, seminar, workshop, etc., activities which are held sometimes inside of the institutes and sometimes in restaurants or other institutes. The categories of context data that we have included in our survey are given in Table 1.

Table 2 shows some sample data from our dataset consisting of users' preferences toward SMS (ignore or attend) according to their contextual information, for instance, their location, situation, SMS sender's category (family, colleague, boss, operator, etc.). Each instance embodies four context values (day, users' location, social situation, and the relationship between SMS sender and receiver) and corresponding users' preferences, e.g., check instantly or ignore (will be checked later, perhaps at a convenient time).

Table 1 Types of context in dataset with value

Types of context in dataset	Value
Day	{Monday, Tuesday, ..., Sunday}
Location	{Home, Office, Outdoor}
Social situation	{Meeting, Sleeping, Seminar, Office-work, Family-time, travelling, shopping}
Social relationship	{Boss, Family, Friend, Colleague, Relative, Operator (like Grameenphone, Airtel, BanglaLink and so on)}
SMS preferences	{Both notification and ringtone, Only notification, No notification and No ringtone}

Table 2 Sample data from survey dataset

Context information	Preference
{Day: Monday, Location: Office, Social Situation: Office work, Social Relationship: Boss}	Both ringtone and notification
{Day: Tuesday, Location: Outdoor, Social Situation: Shopping, Social Relationship: Colleague}	Only notification
{Day: Friday, Location: Home, Social Situation: Sleeping, Social Relationship: Boss}	Both ringtone and notification
{Day: Friday, Location: Home, Social Situation: Sleeping, Social Relationship: Relative}	No ringtone, no notification
{Day: Wednesday, Location: Home, Social Situation: Family-time, Social Relationship: Operator}	No ringtone, no notification

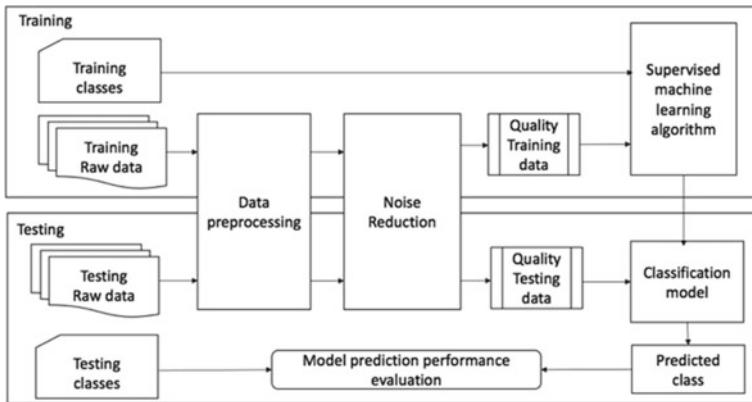


Fig. 1 Block diagram of the proposed system

3.2 Block Diagram

In our proposed model, at first, we detect and remove noisy data from the training dataset by applying the Naïve Bayes Classifier technique. Then we implement machine learning algorithms on the noise-free dataset, i.e., the dataset we round up after removing noise. Here, we implement two machine learning techniques to construct our prediction model: decision tree algorithm and random forest algorithm. The whole press is presented in Fig. 1.

According to the result, we find that the random forest algorithm provides better accuracy than the decision tree algorithm which has been explained in the following part of the paper.

3.3 Dataset Preprocessing

Dataset preprocessing is one of the important steps where null value, missing value, etc., are removed from the dataset. In these datasets all the data are categorical. So, the values are normalized to numerical instances.

3.4 Noise Reduction from Dataset

Input data contains observation data $D = X_1, X_2, \dots, X_n$ which contains training features and classes. The collected dataset may contain noise data which reduces classification accuracy and increase the over-fitting problem. Naive Bayes is a famous classification method to reduce noisy data from the dataset. The class is C and the dataset feature is X in the Naive Bayes method, shown in Eq. (1)

$$P(c/x) = \frac{(P(x/c)P(c))}{P(x)} \quad (1)$$

where

$P(c/x)$ is called the posterior probability of the class given feature.

$P(c)$ is called the class probability.

$P(x/c)$ is called the likelihood which is the probability of a feature given class.

$P(x)$ is called the prior probability of feature in dataset.

Equation (2) is for calculating likelihood for every feature X and Class C in the dataset:

$$p(X|C) = \frac{1}{\sqrt{2\pi} \text{ variance of } X \text{ for } C} e^{-\frac{(\text{observation's } X - \text{ average } X \text{ of } C)^2}{2 \text{ variance of } X \text{ for } C \text{ in the data}}} \quad (2)$$

Equation (3) is for calculating Posterior Probability for every class C :

$$\text{posterior}(C) = \frac{P(C)P(X_1|C)P(X_2|C)P(X_3|C)P(X_4|C)}{\text{marginal Probability}} \quad (3)$$

where

$X_1 = \text{Day}$, $X_2 = \text{Location}$, $X_3 = \text{Social Situation}$, $X_4 = \text{Social Relationship}$.

In the Naive Bayes classifier, the posterior probability has been calculated for every class for each observation. After that, the prediction of the observation has been done based on the largest posterior probability of the class. Then we have found the purely classified and misclassified observations list of the dataset. From the purely classified list, we have found the minimum posterior probability which has been measured as noise threshold for removing noisy data from the misclassified list.

Algorithm 1: Noise detection technique

Data: Training dataset $D=X_1, X_2, \dots, X_n$ which contains training features and classes
 Result: noise list
 for $i=1$ to N
 Find the prior probability of each class $P(C_i)$
 end
 for each feature $f_i \in D$
 for each class $C_i \in D$
 Find Mean $M(f_i/C_i)$ and variance of each $V(f_i/C_i)$
 end
 end
 for each class $C_i \in D$
 find the posterior probability $P(C_i|X_i)$
 if X_i is misclassified then
 misclass_list <- X_i
 else
 pureclass_list <- X_i
 end
 end
 Tnoise = FindMin(pureclass_list)
 for each instance in $X_i \in$ misclass_list
 if $P(C_i|X_i) < Tnoise$ then
 noiselist <- X_i
 end
 return noiselist

3.5 Classification Methods

i. Generating rules by applying decision tree algorithm

In this part, we discuss our prediction model based on decision tree classification algorithm to generate the rules. Figure 2 indicates a pattern of a decision tree for numerous contexts. The values we set right here are mentioned inside Table 2. When the tree has been created, the rules are drawn out through traversing the tree from the root node to each leaf node. The followings are the instances of the introduced rules {R1, R2, ..., R12} constructed from the tree which appeared in Fig. 2.

- R1: Home, Sleeping, mobile operator \Rightarrow no notification and ringtone.
- R2: Home, Family time, Friend \Rightarrow both notification and ringtone.
- R3: Home, personal work/Family time, Relatives \Rightarrow Both Notification & Ring-tone.
- R4: Office, Meeting, Friend \Rightarrow no notification and ringtone.
- R5: Office, Meeting, Boss \Rightarrow both notification and ringtone.
- R6: Office, Office-work, Colleagues \Rightarrow Both Notification & Ringtone.

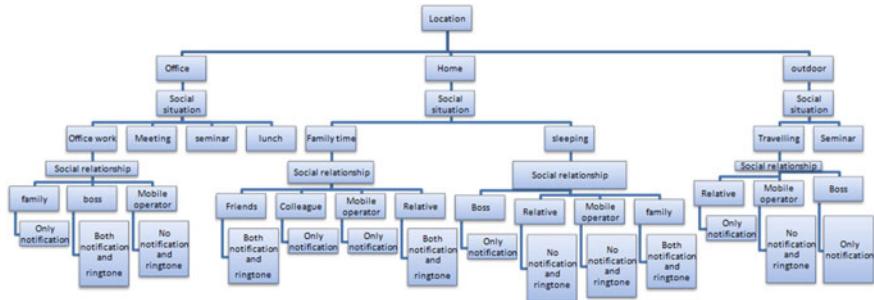


Fig. 2 Decision tree

R7: Office, Office-work, Friends⇒ Only Notification.

R8: Office, office work, Family⇒ Only Notification.

R9: Outdoor, traveling, Boss⇒ only notification.

R10: Outdoor, Seminar, family⇒ only notification.

R11: Outdoor, seminar, mobile Operators = > NO Notification, NO Ringtone.

R12: Outdoor, traveling, mobile operator = > NO notification and ringtone.

ii. Random Forest

A random forest is a collection of decision trees whose results are accumulated into one final result. Their capacity to restrict over-fitting without significantly expanding mistakes because of inclination is the reason they are such ground-breaking models. Decision trees are prone to over-fitting, especially when a tree is particularly profound. Error due to bias and error due to variance can be minimized by random forest.

4 Performance Analysis

In this segment, we explain the results that we obtain from our experiments.

4.1 Assessment Metric

To determine the accuracy of the classification, we match the predicted preferences with the actual preferences and calculate the accuracy in the following terms:

TP denotes true positives, FP denotes false positives, TP denotes true positives and FN denote false negatives then the meaning of precision, Recall, and F-Score are [17]

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

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F - \text{measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.2 Assessment Results

We have utilized the most well-known cross approval procedure, N-fold, in machine learning to assess our model, where we have used fold = 10 to determine the result. In tenfold cross-validation, data are split into ten groups. It makes the prediction model on nine groups analyze its performance utilizing the staying one group. After recurring ten times we have considered the mean accuracy rate. We have coordinated the prediction results of existing methodologies with our models regarding precision, recall, and f-measure to display the effectiveness of our model. Existing methodologies for SMS notification data for modeling without considering the robustness, i.e., the quality of the training data for modeling. For the cause of effectiveness evaluation with our Decision tree robust model and random forest robust model, we show such kinds of existing experiments as ‘Base Model’, which appeared in Fig. 2. we utilize the equivalent datasets portrayed above, in our experimental model to compute fair comparison of the two models, we also utilize the most famous cross approval procedure.

To show the general adequacy as far as to forecast exactness of our powerful model, Fig. 3 shows the overall correlation of accuracy, precision, recall, and f-measure, by figuring the normal outcomes for all the datasets depicted before. On the off chance that we watch Fig. 3, we find that our vigorous model reliably beats the base model for anticipating SMS notification behaviors conduct regarding accuracy, precession, recall, and f-measure. The fundamental explanation is that current base models don’t consider the robustness while anticipating client behavior, and the subsequent precision of both base models is therefore low.

5 Conclusion

In this paper, classification models have been introduced for mobile phone SMS notification preferences focusing on users’ context data to improve the accuracy of the prediction. We have efficiently dealt with the noise data from the training dataset using the Naive Bayes classifier where the posterior probability has been calculated



Fig. 3 Performance analysis

for every class for each observation. After that predicted the observation based on the largest posterior probability of the class and the purely classified and misclassified observations listed in the dataset. From the purely classified list, we obtain the minimum posterior probability which has been measured as noise threshold for removing noisy data from the misclassified list. The well-known rule-based machine learning technique specifically Decision Tree and Random Forest have been implemented on a noise-free quality dataset to construct a complete model. Results from the experiment on mobile phone SMS notification preferences datasets indicate that our proposed model improves the prediction accuracy comparing to the existing approaches in respect of f-measure, recall, and precision.

References

1. Information and Communications in Japan 2015 (summary), 2015|Whitepaper|MIC ICT Policy (2015), https://www.soumu.go.jp/main_sosiki/joho_tsusin/eng/whitepaper/2015/index.html. Accessed 10 Oct 2020
2. Digital 2020: July Global Statshot—DataReportal—Global Digital Insights, DataReportal—Global Digital Insights (2020), <https://datareportal.com/reports/digital-2020-july-global-statshot>. Accessed 03 Oct 2020
3. B. Schilit, N. Adams, R. Want, Context-aware computing applications, in *1994 First Workshop on Mobile Computing Systems and Applications* (1994). <https://doi.org/10.1109/wmcsa.1994.16>
4. T. Kang, A. Moon, R. Kim, H. Kim, H. Cho, Apparatus and method of constructing user behavior pattern based on event log generated from context-aware system environment. U.S. Patent Application 12/058,250 (2009)

5. K. Cheverst, N. Davies, K. Mitchell, A. Friday, C. Efstratiou, Developing a context-aware electronic tourist guide, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems—CHI '00* (2000). <https://doi.org/10.1145/332040.332047>
6. H. Zhu, E. Chen, H. Xiong, K. Yu, H. Cao, J. Tian, Mining mobile user preferences for personalized context-aware recommendation. *ACM Trans. Intell. Syst. Technol.* **5**(4), 1–27 (2015). <https://doi.org/10.1145/2532515>
7. G. Chen, D. Kotz, A survey of context-aware mobile computing research, Technical Report TR2000–381, Dartmouth Computer Science (2000)
8. A. Mukherji, V. Srinivasan, E. Welbourne, Adding intelligence to your mobile device via on-device sequential pattern mining, in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication—UbiComp '14 Adjunct* (2014). <https://doi.org/10.1145/2638728.2641285>
9. A. Mehrotra, R. Hendley, M. Musolesi, PrefMiner: mining user's preferences for intelligent mobile notification management, in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016). <https://doi.org/10.1145/2971648.2971747>
10. I. Sarker, A. Colman, J. Han, RecencyMiner: mining recency-based personalized behavior from contextual smartphone data. *J. Big Data* **6**(1) (2019). <https://doi.org/10.1186/s40537-019-0211-6>
11. K. Fraser, B. Yousuf, O. Conlan, A context-aware, info-bead and fuzzy inference approach to notification management, in *2016 IEEE 7th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference sponsored by IEEE* (Published by IEEE Xplore) (IEEE UEMCON 2016) (New York, USA, 2016)
12. S. Pradhan, L. Qiu, A. Parate, K. Kim, Understanding and managing notifications, in *IEEE INFOCOM 2017—IEEE Conference on Computer Communications* (Atlanta, GA, 2017), pp. 1–9. <https://doi.org/10.1109/INFOCOM.2017.8057231>
13. J. Boase, R. Ling, Measuring mobile phone use: self-report versus log data. *J. Comput.-Mediated Commun.* **18**(4), 508–519 (2013). <https://doi.org/10.1111/jcc4.12021>
14. M. Pielot, K. Church, R. de Oliveira, An in-situ study of mobile phone notifications, in *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services—MobileHCI '14* (2014). <https://doi.org/10.1145/2628363.2628364>
15. M. Pielot, B. Cardoso, K. Katevas, J. Serrà, A. Matic, N. Oliver, Beyond interruptibility, in *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–25 (2017). <https://doi.org/10.1145/3130956>. Accessed 11 Oct 2020
16. I. Sarker, M. Kabir, A. Colman, J. Han, An improved Naive Bayes classifier-based noise detection technique for classifying user phone call behavior, in *Communications in Computer and Information Science*, pp. 72–85 (2018). https://doi.org/10.1007/978-981-13-0292-3_5. Accessed 12 Oct 2020
17. J. Han, J. Pei, M. Kamber, *Data Mining: Concepts and Techniques* (Elsevier, Amsterdam, Netherlands, 2011)

Computer Vision with Deep Learning Techniques for Neurodegenerative Diseases Analysis Using Neuroimaging: A Survey



Richa Vij and Sakshi Arora

Abstract The significant benefits of Computer Vision have enabled computers to gather high-dimensional data from digital images, and videos to make them efficient to act like a human to provide more accuracy. The current literature reveals the superiority of deep learning over traditional method for distinguishing perplexing structures in complex high-dimensional information, especially considering the field of computer vision as well as analyzing medical images mainly brain images or sometimes called neuroimages, acquired via different techniques like Magnetic Resonance Imaging (MRI), Computed Tomography (CT), Positron Emission Tomography (PET), and Single Photon Emission Computed Tomography (SPECT) to diagnose neurodegenerative diseases such as Alzheimer's disease (AD), Parkinson's disease (PD), Huntington's disease (HD), and many others. Such neurodegenerative disorders are a major health care concern worldwide as they are deteriorating people's quality of life to a larger extent. The use of deep learning for early recognition and automated classification of these diseases has attracted the focus of several researchers. This paper surveys the various applications of computer vision in the healthcare field. The key intent of this article is directed towards the concern that how different deep learning architectures can impact and improve the performance of computer vision approaches for more efficient analysis and detection of neurodegenerative diseases.

Keywords Computer vision · Deep learning · Artificial intelligence · Medical imaging modalities · Neuroimages · Neurodegenerative diseases

R. Vij · S. Arora (✉)

School of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra 182320, India

e-mail: sakshi@smvdu.ac.in

R. Vij

e-mail: 19dcs001@smvdu.ac.in

1 Introduction

Computer vision is one of the most remarkable things to emerge from deep learning and artificial intelligence. There are numerous definitions of computer vision, one such was by Prof. Fei-Fei Li and according to his saying Computer vision is a subdivision of artificial intelligence that manages the study of building Personal Computers or machines to see, recognize and process images, the same way as human vision does, and afterward providing a suitable output. Computer vision developments to healthcare provide new insights into Computer-Aided Diagnosis which indeed help doctors in finding discoveries from the medical imaging for disease diagnosis and with the expanding interest for computer vision in the medical industry, the government has taken initiatives to build the usage of computer vision in medicine, other than healthcare some other applications are augmented reality, facial recognition, self-driving cars, and much more as shown in Fig. 1 Computer vision has enlarged remarkable advancement with the increase in the precision rates of object identification and classification. As far as the evolution of computer vision is concerned the first-ever investigation that occurred during the 1950s is to sort the basic objects into two classes and to identify the edges of the object using a small portion of the first neural network. Later in the 1970s, optical character recognition is used to differentiate between typed and handwritten text. With the growth of the internet in the 1990s, face recognition programs succeeded, and by today there is exponential growth in computer vision.

Neurodegenerative illnesses are directly linked to aging and it affects the neuron functioning in the brain which leads to mental impairment and ultimately death and to diagnose these disease, neuroimaging is used which quickest wellspring of clinical information [5]. The present paper performed the vision-based approaches for

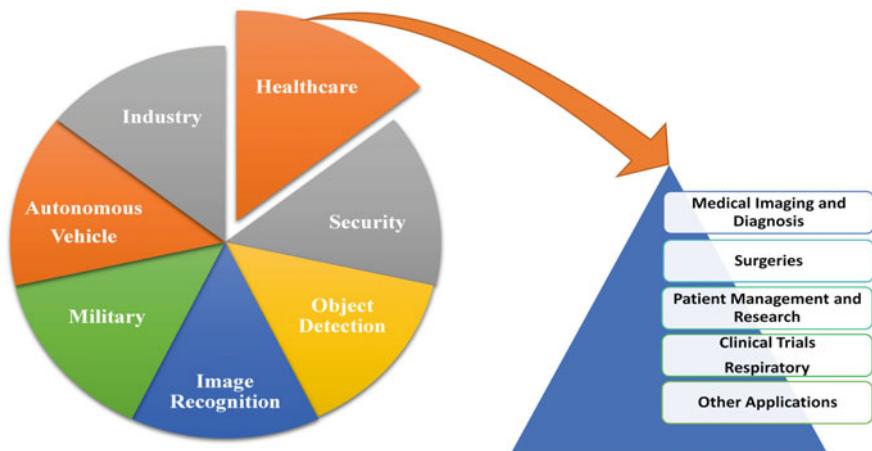


Fig. 1 The pictorial depiction of the most commonly used applications of computer vision emphasizing healthcare applications

the early diagnosis of neurodegenerative disease using medical image analysis. The organization of the paper is as follows: Sect. 2 provides an overview of neurodegenerative disease and its types. Different medical imaging modalities are described in Sect. 3. Section 4 defines the computer vision-based framework in disease diagnosis. Various existing related work will be introduced in Sect. 5. Challenges in the research problem are presented in Sect. 6. Finally, Sect. 7 provides the conclusion and future work of this survey article.

The different terms utilized in this paper are given underneath.

Nomenclature

MRI	Magnetic Resonance Imaging
CT	Computed Tomography
PET	Positron Emission Tomography
SPECT	Single Photon Emission Computed Tomography
DNN	Deep Neural Network
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
SVM	Support Vector Machine
SAE	Stacked Autoencoder
DBN	Deep Belief Network
RBM	Restricted Boltzmann Machine
ANN	Artificial Neural Network
LDA	Linear Discriminant Analysis

2 Neurodegenerative Diseases and Types

Neurodegenerative diseases (NDD) is the umbrella term for so many brain diseases that are described by progressive degeneration of the central nervous system (CNS). As there is no particular cure to this disease and its effect is increased with age. That is why NDD is also called age-related disease [8]. In the early stages, a particular part of the brain is affected thereafter complete impairment happens to that part, later making the brain degenerate. NDD is characterized by Pathological hallmark changes such as collection accumulation of misfolded proteins bringing about intracellular considerations inside neurons [9]. Moreover, there is no evidence of whether the protein leads to the progression of the disease or not [10]. And there is no clarity of how these protein misfolds help in the disease diagnosis. Table 1 shows the pathological hallmarks associated with NDD along with the part of the brain affected.

Out of the various forms of NDD, Alzheimer's (AD), Parkinson's (PD), Huntington's diseases (HD), and multiple sclerosis (MS), consider being the most commonly occurring forms, and according to [11] insights, after every passing 68

Table 1 shows the related protein associated with different brain parts

S.no	> neurodegenerative Disease	Pathological	
1	Alzheimer's Disease	Amyloid- β , Tau	Hippocampus and Cortex
2	Parkinson's Disease	a-Synuclein	Substantia Nigra
3	Huntington's Disease	Huntington	Brain
4	Lewy-body dementia	A Synuclein	Substantia Nigra
5	Amyotrophic Lateral Sclerosis	Superoxide dismutase-1	Mortor Cortex
6	Prion Disease	Prion	Brain
7	Frontotemporal	Tau	Frontal and
f	Dementia		Temporal lobes

seconds, there will be an AD patient. This section provides an overview of neurodegenerative disease and its association with protein mutation and which part of the brain is affected by this.

3 Neuroimaging in Disease Diagnosis

The biggest challenge with the NDD is that there is no test available till now which will evaluate whether the person is experiencing the disease or not but Medical Imaging has given new insights into neurodegenerative diseases by visualizing the internal structures of the brain invasively. There are multiple neuro modalities for taking brain images such as magnetic resonance tomography (MRI), computed tomography (CT), positron emission tomography (PET), and single-photon emission computerized tomography (SPECT). The researcher can choose the imaging accordingly based on the data availability and the purpose of disease diagnosis and there are multiple online sources available for the data and based on the statistics, MRI is considered to be the most used modality for capturing brain images. Below Fig. 2 is the pictorial depiction based on the frequency of usage with time considering the time range from 2004 to 2020.

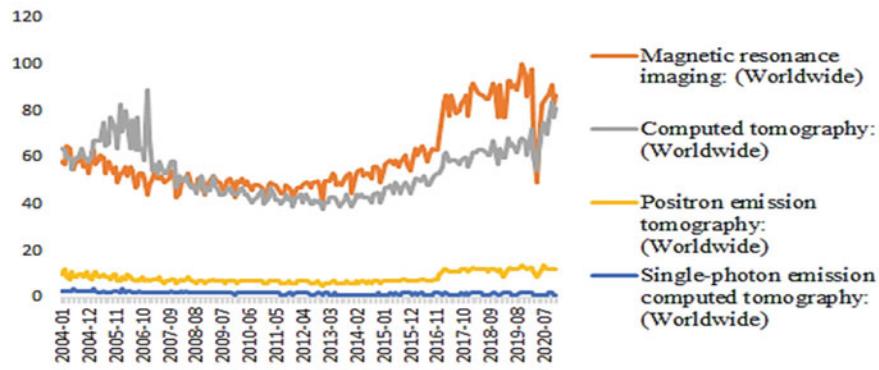
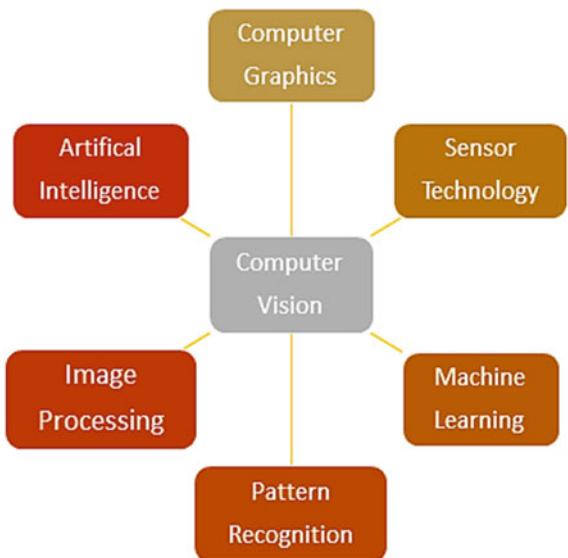


Fig. 2 Representing the stacked line graph showing the usage of different neuroimaging based on the frequency of use by the different researchers and the no of searches in google

4 Computer Vision-Based Framework for Medical Image Analysis

Computer vision is an interdisciplinary field that manages how computers pursue to recognize and mechanize tasks the same as a human visual framework can do [1]. Moreover, it combines two different fields such as artificial intelligence and machine learning. These fields are the subfields of computer vision as described in Fig. 3.

Fig. 3 Computer vision related fields



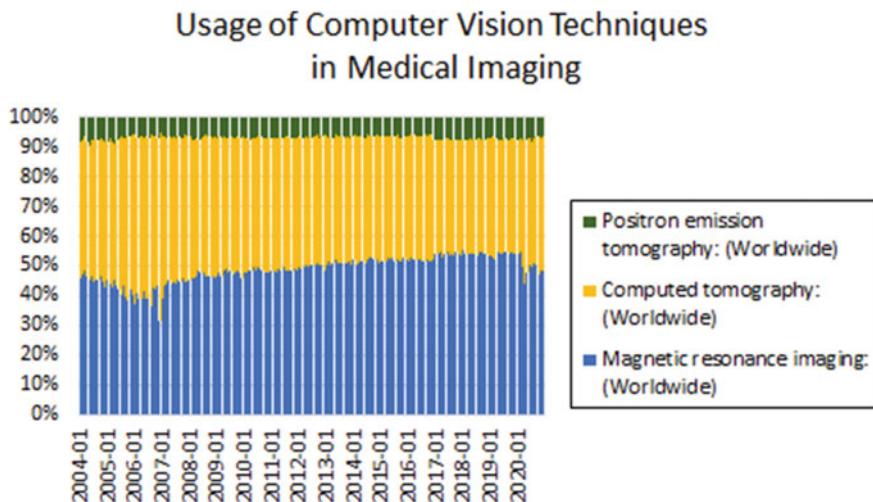


Fig. 4 Pictorial representation of medical imaging used in computer vision techniques for neurodegenerative disease diagnosis

Computer vision is a strategy of artificial intelligence presenting the interpretation of medical images for enhancing diagnostic understanding and information extraction thereby bringing changes in radiology. Particularly in healthcare applications, computer vision makes the identification and diagnosis more precise and prior even before the symptoms may appear making this strategy or technique more efficient [2]. Throughout the most recent years, we have seen the significance of Medical imaging such as magnetic resonance imaging (MRI), computed tomography (CT), positron emission tomography (PET), and so on, for early recognition, analysis, and treatment of illnesses [3]. The advancement in computer vision technologies provides the advancement in medical image analysis and one such analysis is represented in the form of a graph provided by the google trend as shown in Fig. 4.

The essential pipeline for identification of disease incorporates five phases i.e. Data Acquisition followed by pre-processing, feature extraction, classification, and the results these are considered to be the modules of the disease diagnostic system as depicted in Fig. 5 shown below. Imaging sometimes called radiology which is a type of imaging modality that provides the picturing of the interior of the human body that noninvasively helps in diagnosis, monitoring, and treatment of patients with the disease. Generally, medical imaging in neurodegenerative disease is Neuroimaging. The complete pipeline of the diagnostic system is depicted in Fig. 5 showing how the workflows between the different blocks.

At first data, in this case, neuroimaging is acquired from different modality sources as mentioned earlier such as MRI, CT, PET, and SPECT, and this phase is considered as data acquisition. Based on the frequency of publication, every year a huge number of subjects to be precise ten out of thousand are scanned [4]. Once the neuroimaging acquisition is accomplished, pre-processing come into roleplay in which a set of

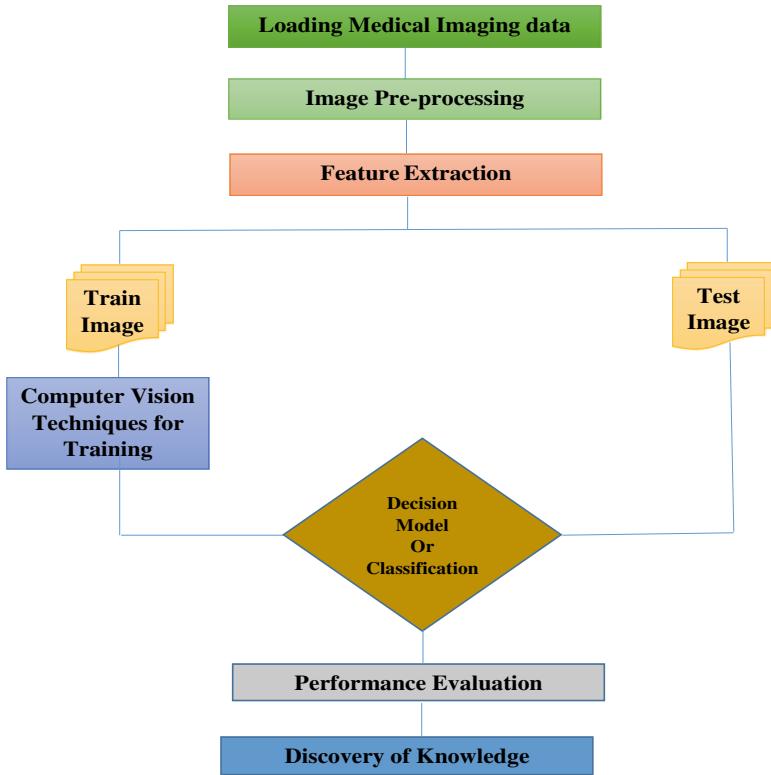


Fig. 5 A pipeline showing the framework of computer-aided diagnostic system

techniques and operations are applied, out of the two methods the frequently used are filtering and segmentation. Once the required data have been processed, the related features are extracted which are then divided into train and test at the ratio of 30:70 and the training data is inputted into the model for the learning phase and test data later used as the input into the model for the validation or classification and at the end performance is evaluated using certain methods and derive some knowledge.

5 Related Work

Various computer vision techniques are applied to the training data. As most of the techniques use neural networks that work like the human brain and whose output is generated when the input is passed through several algorithms [6]. Paliwal and Kumar [7] stated neural networks as incredible computational techniques because of work in the field of physiology and their vast scope in the clinical issue. Among the

Table 2 Different forms of Deep Learning

1	Convolution Networks	Conv-Nets, Res-Net, GoogleNet (AlexNet), U-Net and LeNets
2	Fully Connected Networks	Autoencoders, Convolution Autoencoders, Stacked Autoencoders, Sparse Encoders, and Denoising Autoencoders
3	Belief Networks	Restricted Boltzmann Machine and Deep belief Networks
4	Recurrent Neural Networks	Long Term Short Memory (LSTM), and Gated Recurrent Unit (GRU)

forms of deep learning, CNN and RNN are the popular ones and DNN is one used in computer vision. Table 2. Shows the different types of Deep Learning.

Various theories and methodologies to make the computer recognize and classify like humans have been explained by many researchers in different brain diseases. Table 3 summarizes the related work of the different computer vision techniques in the diagnosis of neurodegenerative disease using medical imaging.

Table 3 Shows the summary of various computer vision-based techniques along with image modalities used for the diagnosis of neurodegenerative disease

Westman et al. [12]	Westman et al. (2011)	AD	MRI	SVM
Wolz et al. [13]	Wolz et al. (2011)	AD	MRI	SVM, LDA
Suk and Shen [14]	Suk and Shen (2013)	AD & MCI	MRI	SAE
Brosch and Tam [15]	Brosch and Tam (2013)	AD	MRI	DBN
Silk et al. [16]	Silk et al. (2014)	AD & MCI	MRI	RBM
Salvatore et al. [17]	Salvatore et al. (2014)	PD	MRI	SVM
Martinez et al. [18]	Martinez et al. (2014)	AD	MRI, PET	SVM
Plis et al. [19]	Plis et al. (2014)	Schizophrenia	MRI	DBN
Payan and Tana [20]	Payan and Tana (2015)	AD & MCI	MRI	CNN
Hosseini-Asi et al. [21]	Hosseini-Asi et al. (2016)	AD & MCI	MRI	CNN
Pinaya et al. [22]	Pinaya et al. (2016)	Schizophrenia	MRI	DBN
Kim et al. [23]	Kim et al. (2016)	Schizophrenia	MRI	ANN
Ortiz et al. [24]	Ortiz et al. (2016)	AD & MCI	MRI	DBN
Shi et al. [25]	Shi et al. (2017)	AD & MCI	MRI	CNN
Ortiz et al. [26]	Ortiz et al. (2017)	PD	MRI	SVM
Choi et al. [27]	Choi et al. (2017)	PD	SPECT	PD-Net
Wang et al. [28]	Wang et al. (2018)	AD	MRI	8L-CNN
Liu et al. [29]	Liu et al. (2018)	AD	MRI	3D-CNN
Huang et al. [30]	Huang et al. (2018)	AD	MRI	CNN
Basaia et al. [31]	Basaia et al. (2019)	AD	MRI	CNN

6 Challenges or Research Questions

Various researchers while working on different methods and dataset encounter some complex challenges which push to develop an efficient method which not only diagnoses neurodegenerative disease with efficiently but also to provide more accuracy than the human expert. Some of the challenges are enlisted below:

- Data availability is always the challenge as health data can't be available online and the one present is incomplete to work on.
- Working on neuroimaging is a little expensive for any financial category of people.
- One weakness to work with DL is that it is hard to modify the potential bias when complexity is high.
- In medical services, data protection is a critical issue because getting the data without influencing their character is challenging for a researcher.

7 Conclusion and Future Work

The rapid growth of people suffering from neurodegenerative disease and the dramatic rise in the treatment cost give rise to the thrust for designing a computer-aided diagnostic system that will overcome all the challenges rising during the research. By making certain efforts, this article surveys the different computer vision with deep learning techniques used in neurodegenerative disease diagnosis by analyzing neuroimaging. Earlier, AI and machine learning are used but with limitations in size and feature extractions, DL has made a big leap to classify medical images. The paper helps to provide an outline of state-of-the-art techniques of DL used in wellbeing areas and problems faced by DL techniques to diagnose neurological diseases. DL is yet advancing to accomplish better performance and it works effortlessly as well in the case of neuroimages but there are some problems with DL which need to be overcome with coming advancement. Moreover, DL will stay functioning in the research for the forthcoming years because of its vast applications in the healthcare sector.

References

1. M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis, and Machine Vision* (Thomson, 2008). ISBN 978-0-495-08252-1
2. A.S. Lundervold, A. Lundervold, An overview of deep learning in ... focusing on MRI. *Z. Med. Phys.* **29**(2), 102–127 (2019)
3. H. Brody, Medical imaging. *Nature* **502**, S81–S81 (2013)
4. K. Smith, Brain imaging: fMRI 2.0. *Nature* **484**, 24–26 (2012)
5. F. Pesapane, M. Codari, F. Sardanelli, Artificial intelligence in medical imaging: threat or opportunity? Radiologists again at the forefront of innovation in medicine. *Eur. Radiol. Exp.* **2**, 35–45 (2018)

6. M. Liu, J. Zhang, E. Adeli, D. Shen, Joint classification and regression via deep multi-task. *Med. Image Anal.* **43**, 157–168 (2018)
7. M. Paliwal, U.A. Kumar, Neural networks and statistical techniques a review of applications. *Expert Syst. Appl.* **36** (2009)
8. W. Noble, D.P. Hanger, C.C.J. Miller, S. Lovestone, The Neurology **80**, 496–503 (2013)
9. J.P. Taylor, J. Hardy, K.H. Fischbeck, Toxic proteins in neurodegenerative disease. *Science* **296**, 1991–1995 (2002)
10. D.M. Hatters, Protein misfolding inside cells: the case of huntingtin **60**(11), 724–728 (2008)
11. Alzheimer's Association, Alzheimer's disease facts and figures. *Alzheimer's Dement. J. Alzheimer's Assoc.* **8**, 131–168 (2012)
12. E. Westman, A. Simmons, Y. Zhang, J.-S. Muehlboeck, C. Tunnard, Y. Liu, L. Collins, A. Evans, P. Mecocci, B. Vellas et al., Multivariate analysis of mri data for alzheimer's disease, mild cognitive impairment and healthy controls. *Neuroimage* **54**(2), 1178–1187 (2011)
13. R. Wolz, V. Jukunen, J. Koikkalainen, E. Niskanen, D.P. Zhang, D. Rueckert, H. Soininen, J. Lötjönen, A.D.N. Initiative et al., Multi-method analysis of MRI images in early diagnostics of Alzheimer's disease, *PloS one* **6**(10), e25446 (2011)
14. H.I. Suk, D. Shen, Deep learning-based feature representation for AD/MCI classification, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, Berlin, Heidelberg, 2013), pp. 583–590
15. T. Brosch, R. Tam, Alzheimer's disease Neuroimaging Initiative. Manifold learning of brain MRIs by deep learning, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, Berlin, Heidelberg, 2013), pp. 633–640
16. H.I. Suk, S.W. Lee, D. Shen, Alzheimer's disease neuroimaging initiative. Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage* **101**, 569–582 (2014)
17. C. Salvatore, A. Cerasa, I. Castiglioni, F. Gallivanone, A. Augimeri, M. Lopez, G. Arabia, M. Morelli, M. Gilardi, A. Quattrone, Machine learning on brain mri datafor differential diagnosis of parkinson's disease and progressive supranuclear palsy. *J. Neurosci. Methods* **222**, 230–237 (2014)
18. F.J. Martnez-Murcia, J.M. Gorri, J. Ramrez, I. Illian, A. Ortiz, P.P.M. Initiative et al., Automatic detection of parkinsonism using significance measures and component analysis in datscan imaging. *Neurocomputing* **126**, 58–70 (2014)
19. S.M. Plis, D.R. Hjelm, R. Salakhutdinov, E.A. Allen, H.J. Bockholt, J.D. Long, V.D. Calhoun, Deep learning for neuroimaging: a validation study. *Front. Neurosci.* **8**, 229 (2014)
20. A. Payan, G. Montana, Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks. *arXiv preprint arXiv:1502.02506* (2015)
21. E. Hosseini-Asl, G. Gimel'farb, A. El-Baz, Alzheimer's disease diagnostics by a deeply supervised adaptable 3D convolutional network. *arXiv preprint arXiv:1607.00556* (2016)
22. W.H. Pinaya, A. Gadelha, O.M. Doyle, C. Noto, A. Zugman, Q. Cordeiro, A.P. Jackowski, R.A. Bressan, J.R. Sato, Using deep belief network modelling to characterize differences in brain morphometry in schizophrenia. *Sci. Rep.* **6**, 38897 (2016)
23. J. Kim, V.D. Calhoun, E. Shim, J.H. Lee, Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance: evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. *Neuroimage* **124**, 127–146 (2016)
24. A. Ortiz, J. Munilla, J.M. Gorri, J. Ramirez, Ensembles of deep learning architectures for the early diagnosis of the Alzheimer's disease. *Int. J. Neural Syst.* **26**(07), 1650025 (2016)
25. J. Shi, X. Zheng, Y. Li, Q. Zhang, S. Ying, Multimodal neuroimaging feature learning with multimodal stacked deep polynomial networks for diagnosis of Alzheimer's disease. *IEEE J. Biomed. Health Inform.* **22**(1), 173–183 (2017)
26. A. Ortiz, F. Lozano, J.M. Gorri, J. Ramirez, F.J. Martinez Murcia, Discriminative sparse features for Alzheimer's disease diagnosis using multimodal image data. *Curr. Alzheimer Res.* **15**(1), 67–79 (2017)

27. H. Choi, S. Ha, H.J. Im, S.H. Paek, D.S Lee, Refining diagnosis of Parkinson's disease with deep learning-based interpretation of dopamine transporter imaging. *NeuroImage: Clin.* **16**, 586–594 (2017)
28. S. Wang, P. Phillips, Y. Sui, B. Liu, M. Yang, H. Cheng, Classification of Alzheimer's disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling, pp. 1–11 (2018)
29. M. Liu, Multi-modality cascaded convolutional neural networks for Alzheimer's Disease diagnosis, pp. 295–308 (2018)
30. Y. Huang, J. Xu, Y. Zhou, T. Tong, X. Zhuang, Diagnosis of Alzheimer's disease via multi-modality 3d convolutional neural network, **13** (2019)
31. S. Basaia, F. Agosta, L. Wagner, E. Canu, G. Magnani, R. Santangelo et al., Automated classification of Alzheimer's disease and mild cognitive impairment using a single MRI and deep neural networks. *NeuroImage Clin.* **21**, 101645 (2019)

Breast Cancer Risk Prediction Using Different Clustering Techniques



Laboni Akter, M. Raihan, Md. Mohsin Sarker Raihan, Mounita Ghosh, Nasif Alvi, and Ferdib-Al-Islam

Abstract Breast Cancer is one of the topmost well-known diseases with a high death rate among women. It is a non-communicable disease that is seen in numerous women in all over the world. With the early analysis of this disease, the endurance will arise from 56% to over 86%. In this analysis, several unsupervised learning techniques were used with the kernel techniques of Principle Component Analysis (PCA). K-Means and several Hierarchical Clustering techniques with different linkages such as ward, complete, and average were applied and highest accuracy of 70.91% was obtained from Hierarchical Clustering with average linkage. The better performances were in Recall and F1-score from K-Means compared to Ward and Complete linkage clustering techniques. The Specificity, Precision, Recall, and F1-score have shown satisfactory performances in Average linkage with the values of 60%, 70.58%, 80%, and 75% correspondingly.

Keywords Breast cancer · K-Means · Hierarchical cluster · Principle component analysis · RBF Kernel

L. Akter · Md. M. S. Raihan · M. Ghosh
Khulna University of Engineering and Technology, Khulna, Bangladesh

M. Raihan · N. Alvi
North Western University, Khulna, Bangladesh
e-mail: rianku11@gmail.com; raihan1146@cseku.ac.bd

N. Alvi
Khulna University, Khulna, Bangladesh

Ferdib-Al-Islam (✉)
Northern University of Business and Technology, Khulna, Bangladesh

1 Introduction

Breast Cancer (BC) is a leading illness that is experienced by numerous ladies worldwide. These cancer rates overall are increasing similarly day by day. The World Health Association showed that breast malignancy impacts 2.1 million women every year [1].

A big illness that is present in many women in India is Breast Cancer. BC is a condition in which cancer cells develop in a woman's breast tissue. The breast consists of lobes (15–20 sections) as well as ducts. In the cells of the ducts, the most prevalent form of breast cancer starts. Cancer starts as other forms of BC in the lobes or lobules present in both breasts. A mark for BC is a warm, red, and swollen breast. The risk of getting breast cancer may be affected by age and health records. BC is caused by variations of chromosomes. To diagnose the stages of Breast Cancer, chest X-rays, CT scans, bone scans, and PET scans are used. Recurrent BC is cancer that returns after therapy is undergone. Cancer can recur in the breast, the upper wall of the body, or any part of the body. This study applied clustering data mining to solve the issue of breast malignancy and early detection of breast cancer [2].

BC is the primary cause of death of females aged 40–55 years and is the second topmost reason of passing after lung cancer. One of the most common cancers in 2012 was BC, according to WHO figures. Annually, more than 1.2 million people worldwide are diagnosed with BC. Fortunately, in recent years, because of the focus on diagnosis and management methods, the mortality rate caused by BC has declined. Quick and right, the key factor in this step is correct prognosis [3].

Machine Learning (ML) is an area of Artificial Intelligence that utilizes factual strategies extensively used in bioinformatics and particularly in BC development end [1]. Two types of learning are mostly found in ML techniques such as unsupervised and supervised. Clustering is one kind of unsupervised learning technique. It is a learning technique which can group similar information in the datasets without knowing the output. In this clustering algorithm, K-Means is most broadly utilized. Now-a-days breast cancer explores utilizing ML strategies primarily based on classification and clustering techniques. Cluster analytical methods are generally used to decide whether newly diagnosed patients with breast cancer can be distinguished on the fundamental of their coping responses as well as to explore the relationships between these coping explanations, decision-making control preferences, and adaptation patterns [4].

The research's objective is to utilize the different clustering techniques to identify the clusters and predict the risk of breast cancer. Different pieces of the original copy are orchestrated as follows: in Sects. 2 and 3, the related works and technique have been expounded. In area 4, the experimented aftereffects of this undertaking have been chaired with the dictation to legitimize the oddity of this investigation. At last, the exploration of this paper is ended with Sect. 5.

2 Related Works

In [4], the significant method clustering was used for determining the patients with breast cancer and to diagnose the relationship among responses, preferences as well as adjustment. They used three clusters for the prediction of breast cancer. The total number of patients were 70 whereas, the number of the active patients were 35 and the number of passive patients were also 35. In [5], they used the cluster analysis method with feature selection for the diagnosis of breast malignancy. The computational coincident measurements were used to explore the salient properties and for exploring clusters. They introduced Instance-Based Nearest and Farthest neighbors (IBNF) as well as three methods of collection of filter-based features: PCA, variance (Var.) with Max-Rel. Using four popular clustering learning techniques: K-means, SOM, HC, and PAM, the chosen features were used to conduct clustering. Output was measured utilizing DBI, CH as well as R-squared validities, which are widely utilized in the study of mathematical models and clusters. In [6], a significant technique was proposed on the basis of the kernel and that was modified from KC-Means which combined Fuzzy C-Means algorithm, K-Means as well as kernel method. The C-Means technique was utilized to the origin of a specific number of groups identified by K-Means, as well as it was assumed that the kernel function could increase the precision of the classification with the ability to distinguish data that could not be separated linearly. The dataset contained 85 samples and the obtained accuracy was 85.26% with fast fuzzy clustering, whereas fast fuzzy clustering based on kernel was 89.74% for the prognosis of breast cancer.

3 Methodology

The overall procedure of the analysis is shown in Fig. 1.

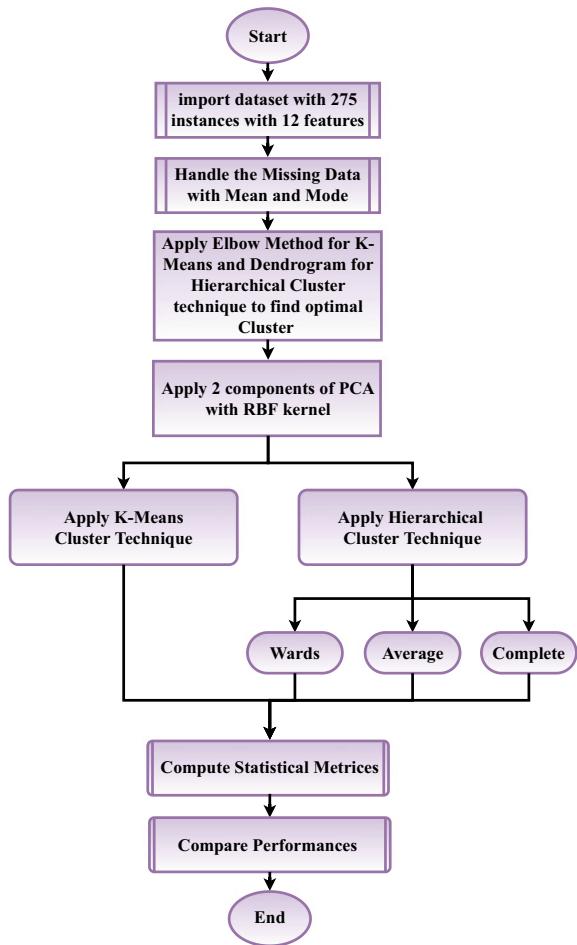
3.1 Instances and Dataset

We accumulate the data from the UCI Machine Learning Repository [7]. This dataset comprised 275 records and 12 features.

3.1.1 Start Age

The feature Start Age has the minimum value of 20 and maximum value of 70. The mean and standard deviation of Start Age were 46.76 and 10.22.

Fig. 1 Workflow of the analysis



3.1.2 End Age

The feature End Age has the minimum value of 29 and maximum value of 79. The mean and standard deviation of End Age were 55.764 and 10.22.

3.1.3 Start Tumor Size

The feature Start tumor size has the minimum value of 0 and maximum value of 50. The mean and standard deviation of Start tumor size were 24.41 and 10.66.

3.1.4 End Tumor Size

The feature End tumor size has the minimum value of 4 and maximum value of 54. The mean and standard deviation of End tumor size were 28.41 and 10.66.

3.1.5 Start_inv_nodes

Start_inv_nodes has the minimum value of 0 and maximum value of 24. The mean and standard deviation of Start_inv_nodes were 1.38 and 3.27.

3.1.6 End_inv_nodes

End_inv_nodes has the minimum value of 4 and maximum value of 26. The mean and standard deviation of End_inv_nodes were 3.38 and 3.27.

3.1.7 Deg-Malig

Deg-malig has the minimum value of 1 and maximum value of 4. The mean and standard deviation of Deg-malig were 2.04 and 0.75.

3.1.8 Menopause

The subcategories of this feature are Premeno, Ge40, and It40 having 142, 126, and 7 instances, respectively.

3.1.9 Node-Caps

This feature contains two subcategories—Yes and No which have 53 and 216 instances correspondingly.

3.1.10 Breast

The subcategories Left and Right contain 148 and 127 instances, respectively.

3.1.11 Breast-Quad

The left_low, left_up, right_low, and right_up have 10, 94, 21, and 33 instances, respectively.

3.1.12 Irradiate

These two subcategories Yes and No have 58 and 217 instances correspondingly.

3.1.13 Class

The No-recurrence class contains 193 and recurrence class contains 82 instances.

3.2 *Missing Data Handling*

To handle the missing information, the mean and mode technique has been applied. The mode is the most time occurring number in a lot of information.

3.3 *Find the Optimal Number of Clusters*

For clustering the dataset, it is very important to obtain the perfect number of K or number of clusters. In this study, the Elbow method with Within Cluster Sum of Squares (WCSS) algorithm have been used for K-Means Clustering and Dendrogram with different kernels (ward, average, complete) for Hierarchical clustering to find out the optimal number of clusters.

3.4 *Elbow Method (WCSS Method)*

Elbow method is one of the widely recognized and actually powerful techniques. Certain target works in K-Means and estimates the distances from their cluster centroids which is called Within Cluster Sum of Squares (WCSS).

3.5 *Dendrogram*

Dendrogram [9] is a visual representation of the compound association data. The individual blends are engineered along the base of the dendrogram and implied as leaf hubs. Compound groups are formed by joining solitary blends or existing compound group with the join point suggested as a hub.

3.6 Applying Principal Component Analysis (PCA) with Kernel

PCA [10] is a broadly utilized procedure for dimensionality decrease of the enormous informational collection. Diminishing the quantity of segments or highlights costs some exactness and then again, it makes the enormous informational collection more straightforward, simple to investigate and picture. Additionally, it diminishes the computational unpredictability of the model which makes Machine Learning calculations run quicker.

3.7 Radial Basis Function Kernel (RBF)

RBF is a kernel function which can be used in ML to find a non-linear classification technique or regression line [10].

3.8 Applying Clustering

In ML, clustering is a method of unsupervised learning and it's a typical procedure for statistical data analysis utilized in numerous areas. Clustering involves the grouping of data points. Clustering technique is to increase the knowledge from data by observing of the bunches of data points. This study has worked with K-Means cluster and Hierarchical cluster for BC dataset.

3.9 K-Means Cluster

The K-Means algorithm clusters information by attempting to isolate tests in n groups of equivalent variances, limiting a measure known as the inactivity or inside group whole of squares. This algorithm requires the quantity of clusters to be indicated. It scales well to huge number of tests and has been utilized over a huge scope of utilization zones in a wide range of fields [11].

To determine K-Means Euclidean distance equation can be used which is as follow [11]:

$$\text{distance}(a, b) = \sum_j^k (a_j - b_j)^2$$

3.10 Hierarchical Clustering (HC)

HC Analysis [11] is a calculation that groups comparative purpose within bunches named clusters. The process of the final stage is very complex. Top-down clustering procedure is generally called divisive. Every bunch that goes recursively till there with this in that is one group for each acumen. At that point decide the resemblance of the clusters with one another and affix the two most similar groups and rehash stages two and three till there is just a solitary group left [11].

3.10.1 Ward Linkage

Ward's strategy [11] is additionally called the base fluctuation technique. Like different calculations, Ward's technique starts with one group for every individual example. At every emphasis, among all sets of clusters, it blends the pair that delivers the littlest squared blunder for the subsequent arrangement of groups. The squared blunder for each bunch is characterized as follows. On the other hand, a bunch contains m tests x_1, \dots, x_m where x_i is the element vector (x_{i1}, \dots, x_{id}) , the squared blunder for test x_i which is the squared Euclidean good ways from the mean is

$$\sum_{j=1}^d (x_{ij} - \mu_j)^2$$

where μ_j is the average value of feature j for the samples in the cluster [11].

3.10.2 Complete Linkage

Separation within 2 clusters is described so there is full separation within 2 focuses in every cluster [11].

$$L(s, t) = \max(D(x_{si}, x_{tj}))$$

The separation within clusters 's' then 't' of the left is equivalent to the extent of the bolt within their 2 farthest focuses.

3.10.3 Average Linkage

Separation within 2 bunches is described as the normal separation within every dot in a single group to each point in the other cluster [11].

$$L(s, t) = \frac{1}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} D(x_{si}, x_{tj})$$

The separation between clusters ‘s’ and ‘t’ to the left is equivalent to the normal length every bolt between interfacing the purposes of one cluster to the next.

4 Experimented Analysis and Discussions

4.1 Accuracy (ACCR)

Accuracy is one estimation for evaluating course of action models. It can be as follows [12]:

$$\text{ACCR} = \frac{\text{TPS} + \text{TNS}}{\text{TPS} + \text{TNS} + \text{FPS} + \text{FNS}}$$

where TPS = True Positives, TNS = True Negatives, FPS = False Positives, and FNS = False Negatives.

4.2 Precision (PRC)

It is defined as the fraction of true positive (TPS) and sum of total positives [12].

$$\text{PRC} = \frac{\text{TPS}}{\text{TPS} + \text{FPS}}$$

4.3 Recall (Sensitivity) (RECL)

It is defined as follow [12]:

$$\text{RECL} = \frac{\text{TPS}}{\text{TPS} + \text{FNS}}$$

4.4 F1-Score

F1-Score is called the harmonic average of precision and recall [12]:

$$F1 - \text{Score} = 2 * \left(\frac{\text{PRC} * \text{REC}}{\text{PRC} + \text{REC}} \right)$$

4.5 Specificity (SPE)

It is defined as correct negative predictions over actual negative values [12].

$$\text{SPE} = \frac{\text{TNS}}{\text{FPS} + \text{TNS}}$$

4.6 Explanation of the Analysis

In Fig. 2, the elbow method and dendrogram showed the optimal number of clusters were three but in the collected dataset, the actual classes were two. We used two clusters instead of three and compared with the real classes. The given chart in Fig. 3 shows a two-dimensional portrayal of the dataset which grouped the entire dataset into two sections utilizing PCA. PCA reduced the higher dimension data into two dimensions.

Table 1 shows the statistical results of this work. The highest accuracy (70.9%) was found in HC with average linkage where the accuracy of K-Means, HC with ward linkage, HC with complete linkage algorithms were 0.65, 0.65, 0.58, and 0.709,

Table 1 Outcome

Evaluation parameters	Algorithm name			
	K-Means	HC_Ward	HC_Complete	HC_Average
Accuracy	0.65455	0.654545	0.581818182	0.709090909
Error	0.34545	0.345455	0.418181818	0.290909091
Specificity	0.51852	0.555556	0.464285714	0.6
Precision	0.62857	0.636364	0.558823529	0.705882353
Recall	0.78571	0.75	0.703703704	0.8
F1-score	0.69841	0.688525	0.62295082	0.75

HC = Hierarchical Clustering Technique

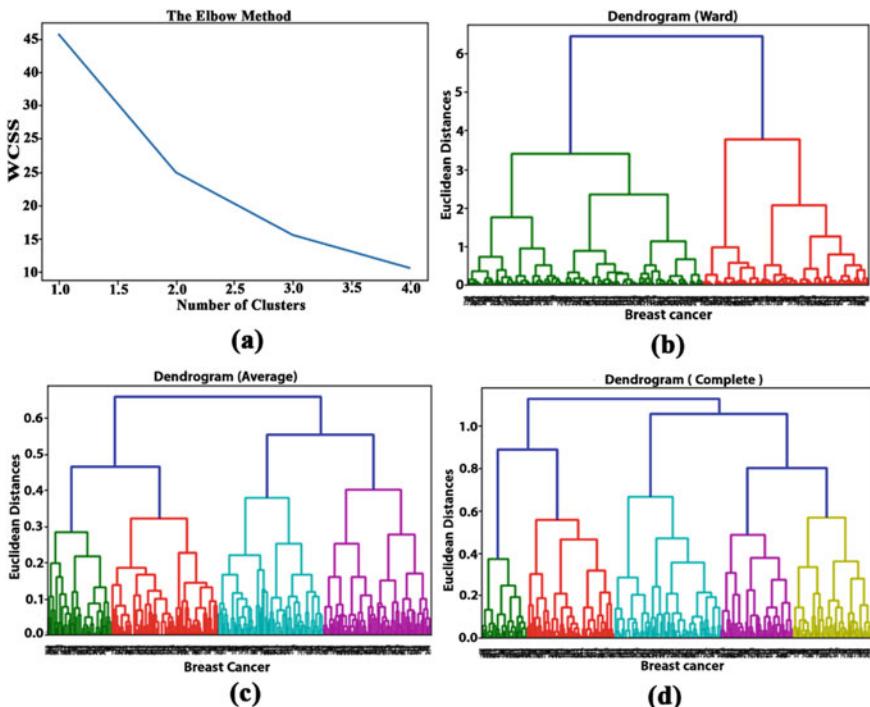


Fig. 2 **a** Elbow method, **b** HC (ward) dendrogram, **c** HC (average) dendrogram, **d** HC (complete) dendrogram

respectively. The error were found 0.345, 0.345, 0.4181 and 0.2909 respectively in K-Means, HC with ward linkage, HC with complete linkage and HC with average linkage. The lowest error was found in HC with average linkage. The maximum specificity, precision, recall, and F1-scores were also found in HC with average linkage. Some papers used K-Means algorithm with different attributes, and various sample size of datasets which included different number of clusters. Table 2 shows some previous studies results including to compare with our models.

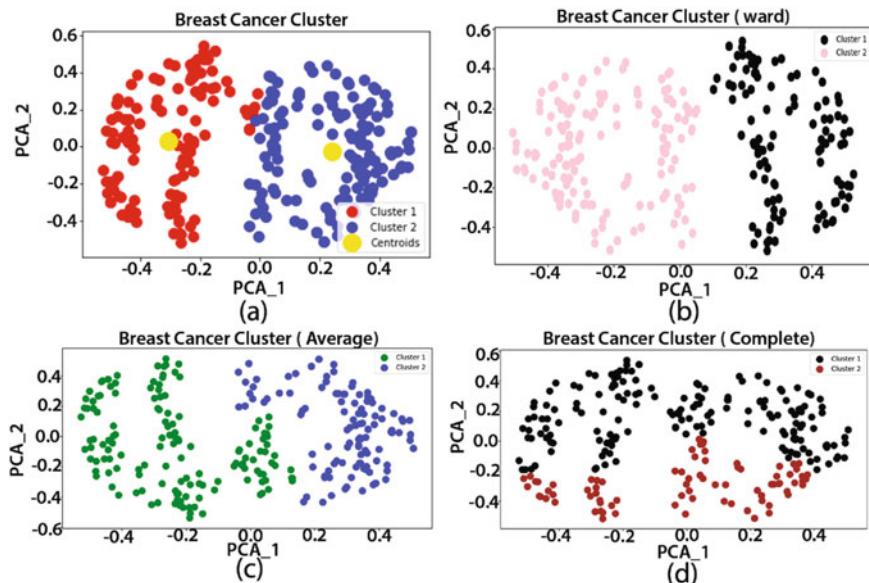


Fig. 3 **a** K-Means cluster, **b** HC (ward) cluster, **c** HC (average) cluster, **d** HC (complete) cluster

Table 2 Comparison with other systems

Reference	Sample size	Number of features	Algorithm used	Number of clusters
Joshi et al. [2]	286	10	K-Means	7
Bahmani et al. [3]	699	11	K-Means, RBF, Naive Bayes	2
Chen [5]	569	32	K-Means, RBF, Naive Bayes	4
Our research study	275	12	K-Means, Wards, Average, Complete	2

5 Conclusion

The proper prediction of breast malignancy gives the chance to be cured of this disease. So, an enormous number of researchers are right now proceeding to find techniques that can identify breast malignant growth more precisely. The outcome of this investigation has shown that K-Means clustering and Hierarchical Clustering are helpful in diagnosing of breast cancer patients. In this study, the optimal number of clusters was also compared with the actual classes. This paper carried out the risk factor of breast cancer with K-Means and Hierarchical Clustering with ward linkage, complete linkage, and average linkage, and the higher accuracy has been found in Hierarchical Clustering with average linkage. Further analysis can be conducted with

other machine learning techniques and an expert system can be developed for the prediction of breast malignancy in females.

References

1. B. Santhosh Kumar, T. Daniya, J. Ajayan, Breast cancer prediction using machine learning algorithms. *Int. J. Adv. Sci. Technol.* **29**(03), 7819–7828 (2020)
2. J. Joshi, R. Doshi, J. Patel, Diagnosis of breast cancer using clustering data mining approach. *Int. J. Comput. Appl.* **101**(10), 13–17 (2014). <https://doi.org/10.5120/17722-7611>
3. E. Bahmani, M. Jamshidi, A.A. Shaltooki, Breast cancer prediction using a hybrid data mining model. *Int. J. Inform. Vis.* **3**(4) (2019). <https://doi.org/10.30630/joiv.3.4.240>
4. T.F. Hack, L.F. Degner, Coping with breast cancer: a cluster analytic approach. *Breast Cancer Res. Treat.* **54**(3), 185–194 (1999). <https://doi.org/10.1023/a:1006145504850>
5. C.H. Chen, A hybrid intelligent model of analyzing clinical breast cancer data using clustering techniques with feature selection. *Appl. Soft Comput.* **20**, 4–14 (2014). <https://doi.org/10.1016/j.asoc.2013.10.024>
6. Z. Rustam, S. Hartini, Classification of breast cancer using fast fuzzy clustering based on kernel, in *IOP Conference Series: Materials Science and Engineering*, vol. 546, p. 052067 (2019). <https://doi.org/10.1088/1757-899x/546/5/052067>
7. S. Kabiraj, M. Raihan, N. Alvi, M. Afrin, L. Akter, S.A. Sohagi, E. Podder, Breast cancer risk prediction using XGBoost and random forest algorithm, in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (Kharagpur, India, 2020), pp. 1–4. <https://doi.org/10.1109/ICCCNT49239.2020.9225451>
8. S. Kabiraj, L. Akter, M. Raihan, N. J. Diba, E. Podder, M.M. Hassan, Prediction of recurrence and non-recurrence events of breast cancer using bagging algorithm, in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (Kharagpur, India, 2020), pp. 1–5. <https://doi.org/10.1109/ICCCNT49239.2020.9225440>
9. N.K. Haneefa, B.M.A. Desai, R. Sarathi, M. Rathinam, Dendrogram based clustering and separation of individual and simultaneously active incipient discharges in transformer insulation, in *2020 International Conference on Signal Processing and Communications (SPCOM)* (Bangalore, India, 2020), pp. 1–5. <https://doi.org/10.1109/SPCOM50965.2020.9179572>
10. J. Wang, Q. Guo, Kernel principal component analysis: radial basis function neural networks based soft-sensor modeling of polymerizing process optimized by cultural differential evolution algorithm. *Instrum. Sci. Technol.* **41**(1), 18–36 (2013). <https://doi.org/10.1080/10739149.2012.710884>
11. E. Gose, R. Johnsonbaugh, S. Jost, *Pattern Recognition and Image Analysis* (Prentice Hall PTR, Upper Saddle River, 1996)
12. J. Han, M. Kamber, J. Pei, *Data Mining* (Elsevier/Morgan Kaufmann, Amsterdam, 2012)

Learner Model of Intelligent Tutoring System Based on Bayesian Network



Rohit B. Kaliwal and Santosh L. Deshpande

Abstract The main importance of the education system is the intelligent tutoring system (ITS). An ITS is a computer system that wants to give immediate and modified instructions or feedback to learners, usually without the intervention of a teacher. In the ITS, the terms of acquiring skills and knowledge use the technology of artificial intelligence to bring a lot of help to the learners. In this process, human instructors are not essential to contribute to the organization, to overcome this problem it has been used as Bayesian Network (BN). The beginner learner model is the heart of an ITS. The level of understanding of the ITS can be greatly improved by using a BN with high self-learning potential to build an ITS for the beginner concept. The fundamental theory of an ITS for the beginner concept will be mainly discussed. Then, at this stage, the elements of impact on the learning method of the learners are studied from the perception of the expertise in the teaching of the beginner, mutual with the state of learning and the characteristics of the beginner. Based on the BN the correct probability interval for learners to answer four classified stages like stage U are $0.8 \sim 1.0$, stage V are $0.7 \sim 0.8$, stage W are $0.4\text{--}0.6$, and stage X are $0.1\text{--}0.3$. The results of the evaluation confirm that the system has a strong analytical capacity. Finally, an ITS for the beginner concept is developed based on the BN. This model can assess the psychological limit of the beginning learner impartially and can conclude the next activity of the beginning learner. Furthermore, the representation is also adapted for e-learning assessment and assessment outcomes are achieved to demonstrate the effectiveness of the modified model.

Keywords Intelligent Tutoring System (ITS) · Learner Model · Bayesian Network (BN) · Assessment · Learner

R. B. Kaliwal (✉) · S. L. Deshpande
Department of CSE, VTU, Belagavi, Karnataka, India

1 Introduction

For years, the twenty-first century has become the era of the information economy. At present, the overwhelming volume of knowledge has a great influence on conventional ways of study. The conventional method of teaching is far from satisfying the demands of the modern era with the rapid expansion of information and technological advancement [1].

Individual learners are an indicator of the professor's ideals since he strives to help each learner reach his or her full potential. Changed preparation can be done with the help of training machines, and the learners who participate can successfully enhance the training class to a greater degree [2]. Classroom training aids are always optimized for the needs of education, from simple training machines to the application and development of the current intelligent ITS. A new field of study, which is intelligent training, has been shaped by the combination of artificial intelligence technology and computer-aided training. The training resources and research results of special and advanced tutors and tutoring experts in the field can be better integrated through intelligent tutoring system (ITS) [3, 4].

The tutor's teaching effectiveness can be enhanced by the learner's learning burden, and the tutor can conduct empirical research, smart decision-making, and systematic, accurate, and personalized tutoring, effectively advancing the learner's academic success while enhancing overall learner excellence. Therefore, research on intelligent tutoring is increasingly on the agenda in order to achieve the highest goal for educators.

In the process of learning and accumulating knowledge, current ITS can meet the basic educational needs of the learner, but many challenges remain. As a result, the intelligent tutoring framework will include effective learning strategies and material based on the learner's learning styles, highlighting the features of individualized instruction, which is currently a research hotspot. Figure 1 illustrates how it can be used. It creates a Bayesian network (BN)-based information tracking model and a classification algorithm-based learner attitude tracking model. For the learner's learning process, the model provides a stimulating, intense, and supervised learning environment.

In order to mobilize learner excitement to learn and achieve better knowledge outcomes, ITS is essential in achieving personalized knowledge. The level of customized services provided by the ITS is directly related to the growth of the learner model. The learner model reflects the knowledge basis of the learner, characteristics of ability, and personality characteristics. The system provides the most appropriate teaching for learners, based on learner models.

In ITS, the study of the learner model mainly focuses on the boundless debate of learning behavior. Techniques such as BNs, fuzzy logic, neural networks, minimal information models, and model tracking are commonly used. Some of these technologies work well, but neural networks are a classic example of how computation can scale up quickly and at a high cost. In this work, in-depth investigations of the learner model in the ITS should be conducted, and a new approach for creating the learner

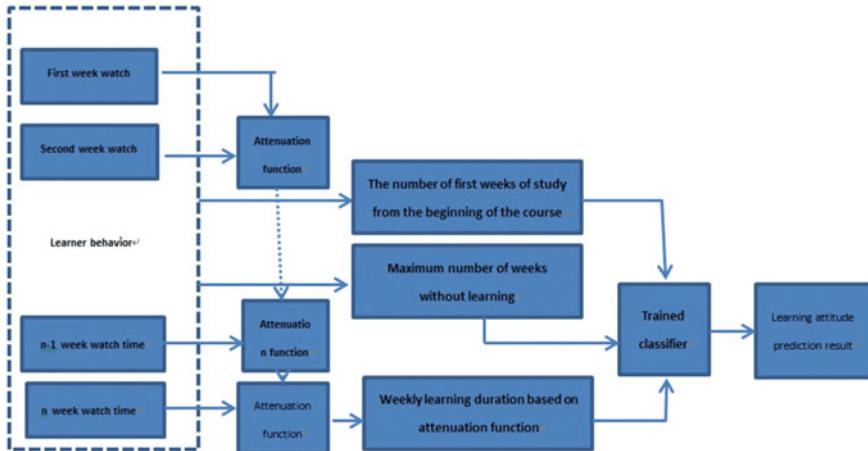


Fig. 1 Knowledge approach prediction model

model with a certain level of logical capacity and a relatively easy implementation will be suggested. This will enhance learning efficiency and pave the groundwork for individualization and intelligence in knowledge/learning systems.

This section is followed by the background work explaining the ITS system. Section 3 explains the—learner model of ITS based on BN followed by the bayesian learner model assessment in Sect. 4, and also it is followed by conclusion and references.

2 Overview of Intelligent Tutoring System

ITS has not been described correctly yet. In normal conditions, the ITS is focused on cognitive science and incorporates artificial intelligence technologies, instructional psychology, computer science, and other multi-disciplinary materials in a holistic manner to provide learners with quality instruction.

ITS has the following advantages compared to conventional training methods [5]: (1) ITS can be learner-centered and completely inspire learners' enthusiasm; (2) ITS can mimic experts' thought processes to build an open collaborative system and shift the conventional style of training; and (3) ITS can conduct the majority of learning and training activities and, to some degree, can improve learner's intelligence [6].

2.1 ITS Mechanisms

In different literatures, the ITS framework is somewhat different. Some experts believe that ITS consist of three parts: the domain knowledge base, the learner, teaching strategies, and reasoning units. According to some experts, the ITS is made up of five components: learner models, teaching methods and inference units, human-machine interfaces (man and machine), expert models, and a field knowledge base [7, 8]. There are four components to this intelligent teaching scheme. The fundamental mechanisms are shown in Fig. 2 between these four parts. Where student = learner.

The domain's knowledge base also includes a cognitive model and an expert knowledge model. It is a component of ITS that is designed in a specific area to solve problems. This domain specifically addresses the issue of "item creation," which includes related information such as definitions, facts, laws, and problem-solving strategies. This model is a standard for assessing learner success or incorrect judgment and is based on expert knowledge [9].

The learner model is the brain of the ITS and is based on the information database. It's in charge of storing the learner's fundamental information as well as dynamic information about the learner's learning process. The learner learning system changes complex data in real time. The learner model is the subject of this research. The teaching strategy and argumentation module receives data from the learner model, chooses the best teaching strategy based on the teaching theory and teaching strategy, and chooses the best teaching material from the base domain's knowledge.

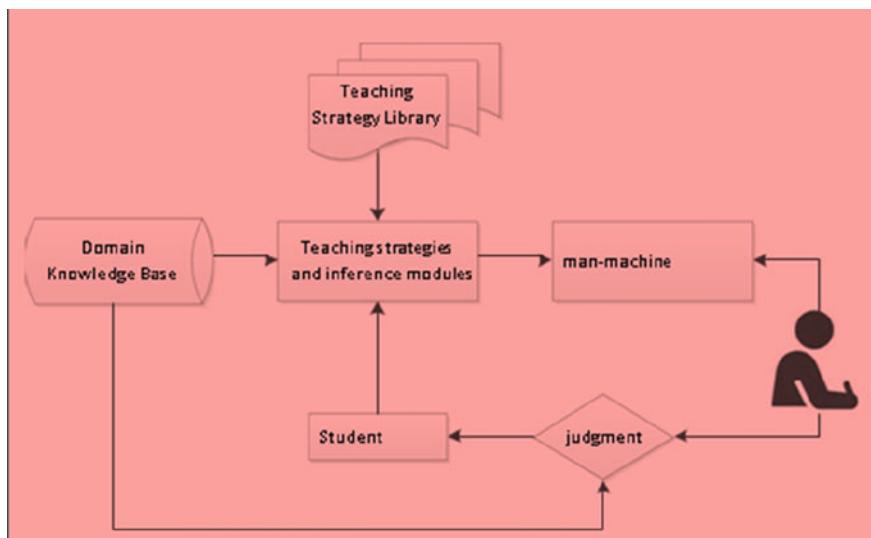


Fig. 2 ITS framework

3 Learner Model of ITS Based on Bayesian Network

Learner model is the main content of the ITS. Based on the learner's response, the system will infer the learner's mastery of various elements of knowledge [9, 10]. The Bayesian method of information representation will aid the learner in comprehending the degree of knowledge he is acquiring. The intelligent tutoring method establishes the domain's information structure through hierarchical knowledge representation, which contributes to knowledge storage, extraction, presentation, and expansion. Figure 3 depicts hierarchical knowledge relations such as inheritance, precursor or succession, parallelism, and connection.

To some degree, a BN may be attributed to the graphic model of artificial intelligence, which often reflects a mixture of graph theory and probability theory [11]. When it comes to simulating causal relationships between artifacts, BNs objects. It is entirely possible to capture the characteristics of prior knowledge and subsequent

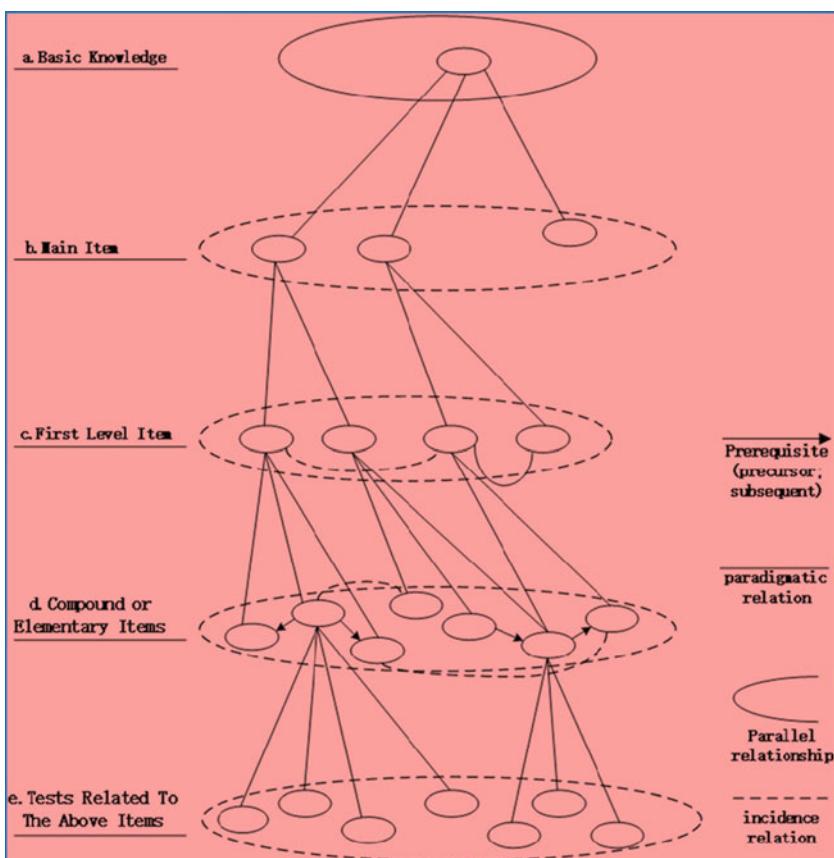
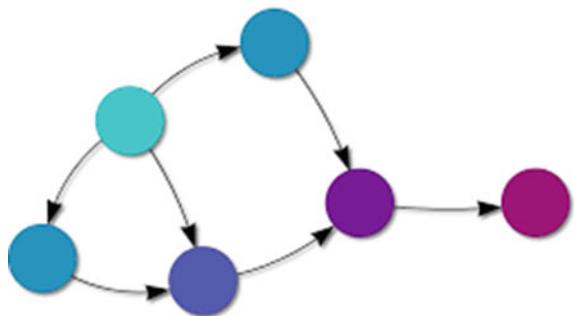


Fig. 3 Knowledge hierarchy graph

Fig. 4 Typical Bayesian network



data when building a learner model, and then compare the practical approximation of the learner's knowledge. The BN thought method is extremely comprehensive, requiring the help of a large number of mathematical theories, and the thinking outcomes are extremely accurate.

3.1 Bayesian Network

The BN is a graph theory-based model that can be represented by a directed no-cyclic graph [9, 11]. It is based on the probability relationship between random variables. In a BN, each variable can be represented by a node (taking A, B, C, D, E, and F as an example). The Bayesian structure of the network is shown in Figure 4. The probability relation between nodes can be represented in a BN by a directed arc and a digitized distribution of probability. The probability relationship of each variable is clearly indicated by the BN structure. Let the variable be x_1, x_2, \dots, x_n and the comprehensive probability when taking the value x_1, x_2, \dots, x_n is

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n \left[\frac{\sum_{i=1}^n P(x_i | X_i) P(X_i)}{\prod_{i=1}^n X_i} \right]$$

where $\frac{\sum_{i=1}^n P(x_i | X_i) P(X_i)}{\prod_{i=1}^n X_i}$ shows the normal probability. The prior likelihood is denoted by $P(X_i)$.

4 Bayesian Learner Model Assessment

In the database management principles course, the definition of “schema” and “definition of sql” are evaluated as information, with KA has “schema definition” and KB has “definition of sql”. Table 1 shows the likelihood of KA and KB meeting. Table 1 shows that KA and KB have a close relationship with dependence; the precise results show that after learning KA, learners can master KB effectively. The assessed scheme is divided into four levels of difficulty: 1, 2, 3, and 4. Discrimination levels are (0, 1), (1, 2), (2, 3), and (4, 5). (3, 4). Table 2 shows the a priori probability distribution of the evaluation difficulty (T: True, F: False). The probability distribution of the frame is shown in Figures 5 and 6 after the evaluation is completed by learners of different levels of knowledge and the network is modified.

From Figures 5 and 6, the X-axis reflects the number of questions occupied in the answer, and Y-axis reflects the likelihood of responding correctly. The correct probability interval for stage U learners to respond is 0.8 ~ 1.0, the correct probability interval for stage V learners’ correct answer is 0.7 ~ 0.8, and the correct probability interval for stage W learners’ correct response is 0.4 ~ 0.6, according to this procedure. For level X students, the proper likelihood interval is 0.1–0.3. The results of the evaluation confirm that the system has a strong analytical capacity.

Table 1 Learning outcome with a conditional probability between KA and KB

KB	KA			
	U	V	W	X
U	0.7232	0.1260	0.0014	0.0011
V	0.1932	0.7743	0.1221	0.1207
W	0.0256	0.1145	0.7632	0.1065
X	0.0026	0.0122	0.1076	0.7723

Table 2 Prior probability distribution calculating the evaluation problem

KA		U	V	W	X
U	T	0.9143	0.8221	0.2623	0.2642
	F	0.0723	0.1243	0.7125	0.7547
V	T	0. 9143	0. 9143	0.4245	0.2642
	F	0.0723	0.0723	0.5545	0.7547
W	T	0. 9143	0.2767	0.7219	0.2642
	F	0.0723	0.5223	0.2767	0.7547

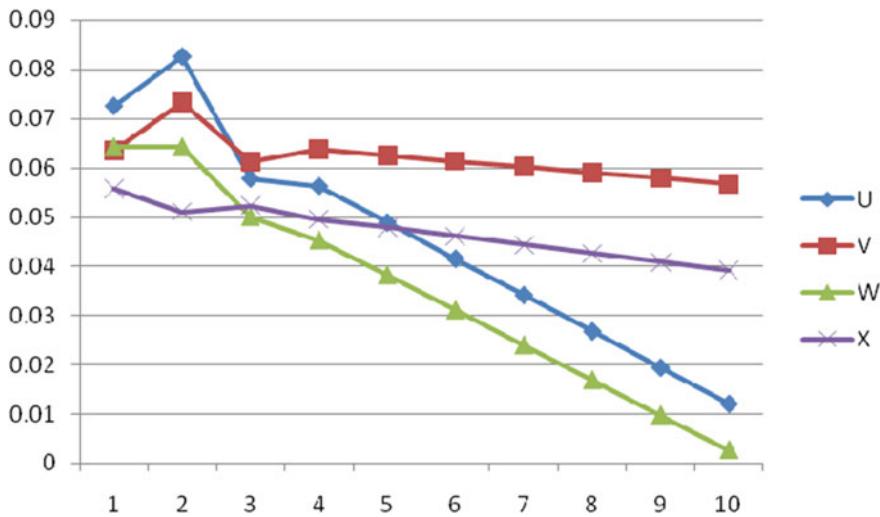


Fig. 5 The learner's probability distribution of KA

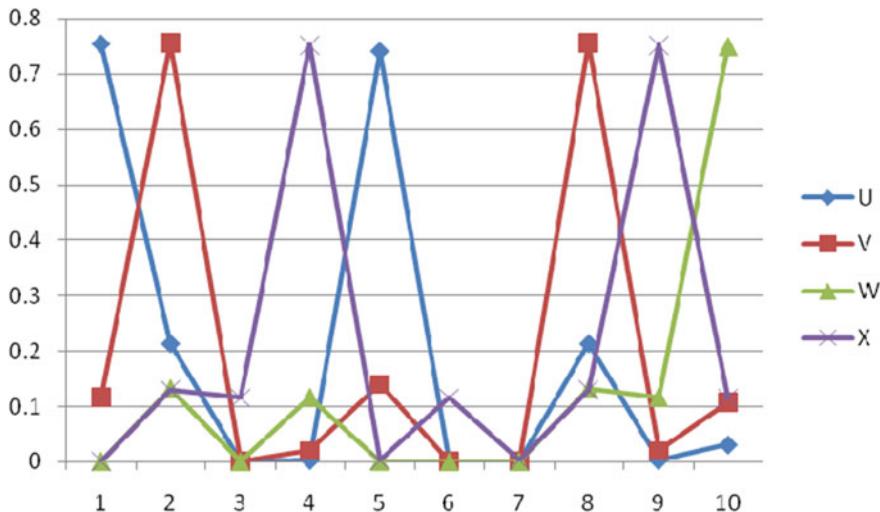


Fig. 6 The learner's probability distribution of KB

5 Conclusion

Multidisciplinary as well as artificial intelligence, diplomat, ontology, internet-based technologies, sharing systems, interactive multimedia, and analysis of data technologies, like an academy, psychology along sociology, are assimilated by the ITS. Its progress is closely linked to computer technology development. This work proposes

a smart learner model for ITS based on BN, a great deal is carried to help beginners learner in requisites of skills as well as knowledge attainment. Based on BN the correct probability interval for learners to answer four classified stages like stage U are 0.8 ~ 1.0, stage V are 0.7 ~ 0.8, stage W are 0.4–0.6, and stage X are 0.1–0.3. The results confirmed that the system has a strong analytical capacity. The learner concept of framework holds a great task of exciting ideas, thorough activity, and supervised learning, as well can ensure individualized training facilities while enhancing the quality of education. The assessment results illustrate the capacity of learner and also eliminate the capacity of learner to deduct the solutions to the queries about the method of learner evaluation and can improve the erroneousness instances.

References

1. W. Zhou, Research on thought and practice of education reform. Jiangsu University (2018)
2. W.R. Nord, *Beyond the Teaching Machine: The Neglected Area of Operant Conditioning in the Theory and Practice of Management* (1969), pp. 375–401
3. M.M. Hilles, S.S.A. Naser, *Knowledge-Based Intelligent Tutoring System for Teaching Mongo Database* (2017), pp. 8783–8794
4. B.X. He, K.J. Zhuang, Research on the intelligent information system for the multimedia teaching equipment management, in *IEEE International Conference on Information System and Artificial Intelligence*, pp. 129–132 (2017)
5. B. Zhang, J. Jia, Evaluating an intelligent tutoring system for personalized math teaching, in *IEEE International Symposium on Educational Technology*, pp. 126–130 (2017)
6. I. Jugo, B. Kovačić, V. Slavuj, Increasing the adaptivity of an intelligent tutoring system with educational data mining: a system overview. *Int. J. Emerg. Techno Learn.*, 423–425 (2016)
7. Y. Bai, *Design of Intelligent Evaluation System of Physical Education Teaching Based on Artificial Intelligence Expert Decision System* (2016), pp. 362–37
8. H. Li, Z. Xu, *Case Study on English Learning Supported by Intelligent Tutoring System* (2013), pp. 66–72
9. R.B. Kaliwal, S.L. Deshpande, Efficiency of probabilistic network model for assessment in e-learning system. *Int. J. Recent Technol. Eng. (IJRTE)* 9(3), 562–566 (2020). ISSN: 2277-3878
10. H. Anderson, M. Koedinger , Intelligent tutoring goes to school in the Big City. *Int. J. Artif. Intell. Educ.* 30–43 (1997)
11. D. Baker, S.J. Ryan, A.T. Corbett, V. Aleven, *More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing* (Intelligent Tutoring Systems, Springer Berlin Heidelberg, 2008)

CADBAIG: Context-Aware Dictionary-Based Automated Insight Generator



Shweta Taneja, Bhawna Suri, Praveen Arora, and Soumya Tanwar

Abstract The backbone of the growth of any field and its contribution toward the society depends on the research and development in that domain. To conduct comprehensive research, the state of art plays an important role but captures a great amount of human hours. In order to reduce this effort, we have proposed an approach to generate insights of research papers. It is done with the help of local context derived from the inherent section level structure of the paper. The local context consists of the title, abstract, and keywords of a research paper. The proposed approach, namely, CADBAIG (Context-Aware Dictionary-Based Automated Insight Generator) consists of six stages. The input to CADBAIG can be the distinct components of local context or the different combinations and the output is the respective extractive summary. The extractive summary is produced by matching multi-word ranked phrases extracted using PyTextRank. The CADBAIG is tested on various research papers. To prove the work, the results are shown by applying the proposed approach on a sample research paper. In addition, to know more about the working of the proposed approach and its respective conclusion, proposed work and conclusion sections of the research paper are also given as input to CADBAIG. The summaries, so generated by the local context, proposed work, and conclusion are compared. It has been observed that the summary generated from local context is more promising than the individual section summaries. The proposed approach can be used to extract the insights from any research paper, irrespective of the domain.

Keywords Natural Language Processing · Insights · Local context · Text Summarization

S. Taneja (✉) · B. Suri · S. Tanwar

Department of Computer Science, Bhagwan Parshuram Institute of Technology, GGSIPU, Delhi, India

e-mail: shwetataneja@bpitindia.com

B. Suri

e-mail: bhawnasuri@bpitindia.com

P. Arora

IPU Affiliated Programmes, JIMS Rohini, Delhi, India

e-mail: praveen@jimsindia.org

1 Introduction

The word Insights in the context of research means interpretation of actual or real data based on observations. Insights of research can be made effective if it has some features like the concept clarity (or unambiguous), easily communicates to the research community about the particular topic, presents key actionable ideas, etc.

The first step of every research is a thorough exhaustive literature review which is a significant contribution to the research community working in that area. This activity is very time-consuming and needs more effort to extract the contribution done by others in that domain. Researchers often conduct an extensive review to the best of their abilities. Extraction or generation of insights from research papers aims at finding the key actionable ideas that can help the researcher as well as the reviewer, to manage and make the research paper more accessible to all by improving the experience of the most difficult stage of a research effort. In this paper, our work focuses on the extraction of these insights from scientific research papers.

While there has been a significant recent interest in automated summarization of scientific research papers, as in the work of authors [1–3], the focus has been on neural network-based methods which are not only data-hungry but also data-dependent. To the best of our knowledge, most of the work is done on research papers belonging to the biomedical or computer science domain.

In this paper, the task of extracting insights from research papers is formulated as an extractive summarizer model with two modules—one for context building and one for ranking sentences to be included in the insights. The context building module considers the inherent relationships between different sections of the paper to identify key terms and score them according to their relative importance in the target representative summary of the paper. This is done by capturing the contextual importance of key terms at a local level. The importance of a key term at the local level is determined by combining the terms that occur in the Abstract, Keywords, and Title section of the paper based on two factors—the frequency occurrence and priority which is set according to how many sections the term appears in. On the local level—key terms are single-word tokens which have an associated score. Further, PyTextRank [4] is used to identify multi-word key terms that extract top-ranked phrases from the target summary source. These multi-word key terms are then further tokenized and assigned normalized scores which are discussed in detail in the Proposed Method section of this paper. Once scoring has been done on the local context, the high-priority keys which contain tokens from locally important words are assigned with the scores. These high-priority keys are then used to score sentences from the target summary source to get the actionable insights of the paper.

The organization of the paper is as follows: Sect. 2 gives the related work done by different researchers in this area. In Sect. 3, the proposed methodology is given. It consists of six stages which are explained in detail. In the next section, working of the proposed methodology is explained with the help of a sample research paper. Section 5 shows the results and comparison part. This is followed by the conclusion and future work section at the end.

2 Related Work

There are a variety of methodologies used in text summarization. Most common among them are graph-based, machine learning-based, statistical-based, deep learning-based, etc.

Many authors are using graph-based methods to infer summaries. In [5], authors have used frequency itemset mining technique to implement graph-based summarization in the bio medical domain. They have created a concept-based model of the text document and then mapped the document with the concepts. Further the authors have used a clustering algorithm to identify sub-topics within the document. In another work [6], authors have proposed a graph-based method named as Edge-Summ that produces a summary for single text documents. It uses four algorithms, in which a new text graph model is designed for the input document, then sentences are searched for generating the summary and finally most important sentences are selected.

Machine learning-based approaches are also being used in this area. In [7], authors have proposed a machine learning-based approach to present users opinions from reviews. They have assigned a sentiment score to each sentence and then used word embedding model to find semantic relationships among words. The authors in [8] have used machine learning methods to detect emotions from emails. They have used three classification algorithms and three feature selection methods to identify emotions. The researchers have also used deep learning-based methods in text summarization. In [9], authors have given a novel method based on deep learning for generating opinion-based text summarization. The method uses sentiment analysis algorithms. Unlike phrase embedding built using machine learning methods that are trained on large corpuses as explored in [10, 11], the graph-based topic modeling techniques are also discussed in [12, 13].

Along with automatic summarization of text documents, another area in which a lot of work is going in is text categorization. The authors in [14] and [15] have proposed a method for multi-label categorization of text documents. It is based on the ideas of lexical and semantics of natural language. In today's era, short text summarization of news articles is a well-researched area but in the domain of research papers the summarization is limited to recommendation search engines [16] and mining of citation networks [17, 18] at large. Insight extraction from research papers is a relatively less explored topic.

3 Proposed Approach: CADBAIG (Context-Aware Dictionary-Based Automated Insight Generator)

To extract insight from a research paper, focus would be on the sections that convey inference of the paper. These could be the Title or Abstract or Keywords or Title + + Keywords + Abstract or Proposed work or conclusion. Considering this fact the

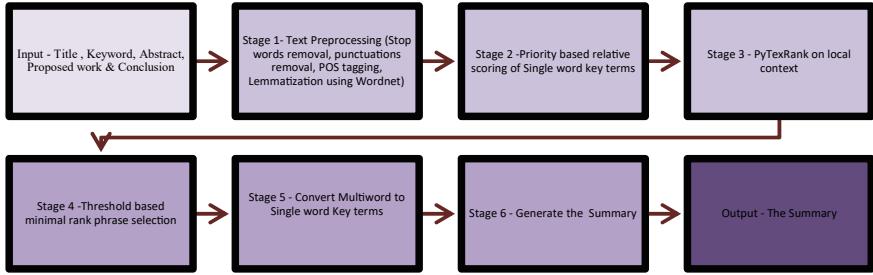


Fig. 1 Architecture of the proposed approach CADBAIG

proposed summary generator CADBAIG traverses through six stages to conclude the summary of the paper without much effort. The detailed architecture of CADBAIG is shown in Fig. 1.

In the first stage, we leverage the structural dependency and overlap of words (tokens) across sections of a paper by identifying single-word key terms in the Title, Abstract, and Keyword sections, respectively. In the second stage, PyTextRank implementation of TextRank is applied to extract the top-ranked multi-word key terms. In the third stage, the minimal ranked phrases extracted from PyTextRank implementation are then selected based on a percentile threshold since the actual value of scores and their relative value may differ for each paper and cannot be known beforehand. The threshold chosen is the phrases with a score above the 50th percentile for that paper's scored key phrases to be selected as phrase candidates. In the fourth stage, we convert these phrases into single-word relatively ranked keywords, so that they can be combined with the single-word scored key terms obtained in the first stage. To remove redundancy and combine words in different from different topics, WordNet-based lemmatization via NLTK is used. In the 5th stage, the lemmatized token dictionary of sectional single-word keys and lemmatized token dictionary of relatively ranked multi-word phrases which have been mapped to single-words is then combined and their respective scores are added to form a dictionary of high-priority keys which are then used to score sentences in summary source and the quality summary which could then be filtered based on the desirable length or percentile. In the sixth stage, after combining the scored keys into the high-priority keys, the sentences in the target summary source are scored to select the most important sentences as insights of the paper using the function. In the following section, the paper on which the implementation of the approach is done is shown.

4 Working of CADBAIG on a Sample Research Paper

To explain the working of CADBAIG, we have taken a sample research paper [19] as an input. Figure 2 shows the sample research paper—Title, Keywords, and Abstract.

Title	Application of balancing techniques with ensemble approach for credit card fraud detection
Keywords	Fraud detection, balancing techniques, SMOTE, Synthetic Minority Over-sampling Technique, ADASYN, Adaptive Synthetic Sampling
Abstract	<p>Fraud detection is an important application area of data mining. Due to the development in technology, there is an increase in the number of frauds nowadays. Detecting the frauds is one of the greatest challenges in organizations. A major challenge that comes while handling frauds in datasets is that the datasets are highly imbalanced in nature. The fraud instances are below 1% as compared to normal transactions. In this paper, our focus is on the comparison of various available balancing techniques in conjunction with classifiers and identified the best combination. The dataset taken is the standard credit card data of a European bank. We have applied different balancing techniques like Down Sampling, Up Sampling, Regular SMOTE, Borderline SMOTE, SVM SMOTE and ADASYN. For comparison purpose, we have taken classifiers as bagging and boosting models. Results have shown that balancing dataset using SVM SMOTE followed by Random Forest classifier gave the best results with F-score value of 0.85.</p>

Fig. 2 Sample research paper given as input to CADBAIG

5 Results and Comparative Analysis

The proposed approach CADBAIG has been tested on many research papers, to show the results we have used the sample research paper as shown in Fig. 2. The extractive summary is produced by matching multi-word ranked phrases extracted using PyTextRank. For the purpose of evaluation we take up to 50th percentile scored sentences as part of the final summary of the method. Figures 3, 4, 5, and 6 show the summary sentences ranked according to their score from Abstract, Title + Abstract + Keyword, Proposed work, and Conclusion.

In the following Table 1 only the summary from Title + Abstract + Keywords is shown which is obtained using CABDAIG. It is sorted in the order of sentences in the source document for the convenient readability and understandability of the user.

Fig. 3 Ranked summary only from abstract

		index		sent	sent_score
0	9	For comparison purpose, we have taken classifi...		0.198584	
1	8	We have applied different balancing techniques...		0.186836	
2	4	A major challenge that comes while handling fr...		0.169089	
3	6	In this paper, our focus is on the comparison ...		0.155149	
4	2	Due to the development in technology, there is...		0.143944	
5	1	Fraud detection is an important application ar...		0.141776	
6	10	Results have shown that balancing dataset usin...		0.123563	
7	3	Detecting the frauds is one of the greatest ch...		0.086353	
8	5	The fraud instances are below 1% as compared t...		0.074201	
9	7	The dataset taken is the standard credit card ...		0.068284	

index			sent	sent_score
0	12	Application of balancing techniques with ensemble models	0.378843	
1	8	We have applied different balancing techniques	0.212418	
2	9	For comparison purpose, we have taken classification	0.184893	
3	6	In this paper, our focus is on the comparison	0.169635	
4	4	A major challenge that comes while handling fraud detection	0.162729	
5	2	Due to the development in technology, there is a need for	0.142925	
6	1	Fraud detection is an important application area	0.129164	
7	11	Fraud detection, balancing techniques, SMOTE, etc.	0.098026	
8	3	Detecting the frauds is one of the greatest challenges	0.088492	
9	5	The fraud instances are below 1% as compared to the total	0.070132	
10	7	The dataset taken is the standard credit card dataset	0.067736	
11	10	Results have shown that balancing dataset using SMOTE	0.059793	

Fig. 4 Ranked summary from title+

index			sent	sent_score
0	7	Lastly, results were compared on the basis of different metrics	0.249821	
1	5	After balancing the dataset, ensemble models were used	0.172884	
2	2	PCA transformation is applied to raw data and features	0.090193	
3	4	Further, the dataset was balanced using different techniques	0.081074	
4	1	Fig.1 shows the flowchart of our proposed work	0.000000	
5	3	The dataset is highly imbalanced	0.000000	
6	6	Models used are Random Forest (RF), Extreme Gradient Boosting (XGB)	0.000000	

Fig. 5 Ranked summary from proposed work

index			sent	sent_score
0	1	Balancing datasets using different techniques	0.198164	
1	3	For comparison purpose, we have taken classification	0.191054	
2	5	Further, results can be improved using hyper-parameter tuning	0.188593	
3	2	In this paper, we have applied different balancing techniques	0.122405	
4	4	Results have shown that balancing dataset using SMOTE	0.000000	

Fig. 6 Ranked summary only from conclusion

6 Conclusion and Future Work

To extract the summary from a given research paper, the different attributes which are the respective sections of the paper will contribute toward it. These different attributes can independently also generate the summary. In addition, if collaborated together, they can also generate the insight in a better and efficient manner. In our

Table 1 Target summary Source Abstract + Keywords + Title

Index	Sentence
1	Fraud detection is an important application area of data mining
2	Due to the development in technology, there is an increase in the number of frauds nowadays
3	Detecting the frauds is one of the greatest challenges in organizations
4	A major challenge that comes while handling frauds in datasets is that the datasets are highly imbalance in nature
5	The fraud instances are below 1% as compared to normal transactions
6	In this paper, our focus is on the comparison of various available balancing techniques in conjunction with classifiers and identified the best combination
7	The dataset taken is the standard credit card data of a European bank
8	We have applied different balancing techniques like Down Sampling, Up Sampling, Regular SMOTE, Borderline SMOTE, SVM SMOTE and ADASYN
9	For comparison purpose, we have taken classifiers as bagging and boosting models
10	Results have shown that balancing dataset using SVM SMOTE followed by Random Forest classifier gave the best results with F-score value of 0.85
11	Fraud detection, balancing techniques, SMOTE, Synthetic Minority Over-sampling Technique, ADASYN, Adaptive Synthetic Sampling
12	Application of balancing techniques with ensemble approach for credit card fraud detection

work, we have taken the different attributes such as title, abstract, keywords, proposed work, and conclusion. The results are produced by taking the above-listed attributes independently as well as in collaboration. We have observed that title, abstract, and keywords together form the local context of the paper and are helpful for generating the insight of a research paper in comparison to handling them independently. The summary so produced does not tell about the approach of the researcher to handle the problem which is clearly explained in the proposed section. In future, to know about the approach, we will consider the proposed work section and the conclusion section together to generate a better and effective insight of a research paper.

References

1. E. Collins, I. Augenstein, S. Riedel, A supervised approach to extractive summarisation of scientific papers. arXiv preprint [arXiv:1706.03946](https://arxiv.org/abs/1706.03946) (2017)
2. M. Nikolov, I. Nikola, M. Pfeiffer, R.H. Hahnloser, Data-driven summarization of scientific articles. arXiv preprint [arXiv:1804.08875](https://arxiv.org/abs/1804.08875) (2018)
3. W. Xiao, G. Carenini, Extractive summarization of long documents by combining global and local context. arXiv preprint [arXiv:1909.08089](https://arxiv.org/abs/1909.08089) (2019)
4. P. Nathan, PyTextRank, a Python implementation of TextRank for phrase extraction and summarization of text documents, <https://github.com/DerwenAI/pytextrank/>. Accessed 21 Nov 2020

5. M.N. Azadani, N. Ghadiri, E. Davoodijam, Graph-based biomedical text summarization: an itemset mining and sentence clustering approach. *J. Biomed. Inform.* **84**, 42–58 (2018)
6. W.S. El-Kassas, C.R. Salama, A.A. Rafea, H.K. Mohamed, EdgeSumm: graph-based framework for automatic text summarization. *Inf. Process. Manage.* **57**(6), 102264 (2020)
7. A. Abdi, S.M. Shamsuddin, S. Hasan, J. Piran, Machine learning-based multi-documents sentiment-oriented summarization using linguistic treatment. *Expert Syst. Appl.* **109**, 66–85 (2018)
8. Z. Halim, M. Waqar, M. Tahir, A machine learning-based investigation utilizing the in-text features for the identification of dominant emotion in an email. *Knowl.-Based Syst.* **208**, 106443 (2020)
9. A. Abdi, S. Hasan, S.M. Shamsuddin, N. Idris, J. Piran, A hybrid deep learning architecture for opinion-oriented multi-document summarization based on multi-feature fusion. *Knowl.-Based Syst.* **106658** (2020)
10. D. Mahata, et al., Theme-weighted ranking of keywords from text documents using phrase embeddings, in *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)* (IEEE, 2018)
11. M. Debanjan, et al., Key2vec: automatic ranked keyphrase extraction from scientific articles using phrase embeddings, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 2 (Short Papers) (2018)
12. R. Mihalcea, P. Tarau, Textrank: bringing order into text, in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing* (2004)
13. A. Bougouin, F. Boudin, B. Daille, Topicrank: graph-based topic ranking for keyphrase extraction (2013)
14. R. Jindal, S. Taneja, A lexical-semantics-based method for multi label text categorisation using word net. *Int. J. Data Mining, Model. Manage.* **9**(4), 340–360 (2017)
15. R. Jindal, S. Shweta, A modified knowledge discovery process in the text documents. *Int. J. Innov. Comput., Inf. Control* **14**(3), 817–832 (2018). ISSN 1349-4198
16. C. Pan, W. Li, Research paper recommendation with topic analysis, in *2010 International Conference on Computer Design and Applications*, vol. 4 (IEEE, 2010)
17. X. Liu, et al., Full-text based context-rich heterogeneous network mining approach for citation recommendation, in *IEEE/ACM Joint Conference on Digital Libraries* (IEEE, 2014)
18. Y. Jia, L. Qu, Improve the performance of link prediction methods in citation network by using H-index, in *2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)* (IEEE, 2016)
19. S. Taneja, B. Suri, C. Kothari, Application of balancing techniques with ensemble approach for credit card fraud detection, in *2019 International Conference on Computing, Power and Communication Technologies (GUCON)* (IEEE, 2019)

Human Depression Prediction Using Association Rule Mining Technique



Md. Al-Mamun Biilah, M. Raihan, Tamanna Akter, Nasif Alvi,
Nusrat Jahan Bristy, and Hasin Rehana

Abstract Depression or Stress is one of the most prominent psychiatric illnesses. It is a common mental neurological disease. It can be distinct as feelings of sadness, loss, annoy which hamper the daily behavior of a person. People undergo various forms of depression. It can interfere with everyday work. In this analysis, we have tried to figure out the most critical depressive factors. The correlation between the factors has been observed. This research study will help us to understand the reason for stress. The dataset for this study has collected from the university students that contains 539 instances where each instance have 23 questionnaires or features. In this study, the 18 most significant features have been identified. Out of these, 9 features are identified as highly correlated, 4 features as medium, and 5 features as lower significant features. In this study, Chi-Square Test have has used for finding the correlation. Association rule mining techniques have been used to get the significant rules. Apriori algorithm has been used to utilize association rule mining and obtained 8 significant rules.

Keywords Depression · Prediction · Human behaviors · Activities · Association rule mining · Apriori · Data mining · Chi-square · Correlation · Asymptotic significance

1 Introduction

The rate of depression or stress is growing day by day. It has become the largest disease in the world that presents a major threat to human health and mind. According to World Health Organization, around 150 million people exist in the world with depressive disorders [1]. Many typical symptoms are the characteristics of

Md. A.-M. Biilah · M. Raihan (✉) · T. Akter · N. Alvi · N. J. Bristy
North Western University, Khulna, Bangladesh
e-mail: raihanbme@gmail.com; raihan1146@cseku.ac.bd

H. Rehana
Rajshahi University of Engineering and Technology, Rajshahi, Bangladesh

depression. Those involve a constant gloomy depressed or soft mood and a sense of hopelessness or pessimism that persist for weeks on end, almost every day. A person who is depressed often has feelings of rumors, worthlessness. While high-quality remedies for intellectual disabilities are common, between 76 percent and 85 percent of people in international locations with low and middle incomes do not take action for their disease [2]. Manpower is a blessing for a country, where depression is one of the biggest obstacles for the country. Depression or stress is a hindrance to people's daily work. It is harmful to the human body and mind together. The suicide rate is 1.8, which means the rate of suicide in males is more than in females [3]. Older adults may also have trouble recognizing cognitive changes because the symptoms of depression may be easily discounted as being linked to getting older. Depression hampers our daily activities.

The motivation of this paper is to explore (a) depression stress level (high, medium, low) based on our daily activities and (b) association among reasons for depression or stress. The goal of this research study is to classify the depression level with the collected data. This research paper's objective is to identify the major reasons for depression or stress.

The rest section of this research study is organized as follows: in Sects. 2 and 3 the existing works and working procedure have been explained with a thorough analysis of the algorithms, respectively. In Sect. 4 the outcome of this analysis has been clarified with the impulsion to justify the novelty of this exploration work. Finally, this research paper is terminated with Sect. 5.

2 Existing Works

By using a multi-modal fusion framework a research paper was published. They proposed this framework which considers auditory, cartridge, and text streams. Their root mean squared error was 4.653 and mean absolute error was 3.980 [4]. In this study, they used the Extreme Gradient Boosting (XGBoost) algorithm to classify the causes of depression and the biomarker's predictability in the diagnosis. In this research paper, they trained on an 11,081 Dutch citizen dataset. They found the best result. The accuracy rate was 0.9729. They found the relationship between self-reported depression and a set of biomarkers [5]. By using Natural Language Processing (NLP) research was done for analyzing emotion on depression. They used a support vector machine and Naive-Bayes classifier for predicting class. They collected data from Twitter. Also, they published results using primary classification metrics including F1-score, confusion matrix, and accuracy [6]. By analyzing social media data another paper was established. They worked with a random forest algorithm. They used two methods of Machine Learning. There one will detect the depressed subjects and another algorithm will help to identify non-depressed individuals. They found that the dual model was more helpful than one [7]. In another paper, the researcher worked on Twitter data for predicting and another hand they compared his result with other papers [8].

3 Methodology

Few comparable works have been carried out for depression analysis and classified the depression into several classes. These works stand as a guideline for our work. This section briefly discusses some techniques and the implementation of efficient algorithms and tools. For this ethical consideration, a survey on record series has been taken to gather statistics from college students from a number of universities in Bangladesh. There are 539 instances in the dataset. We listed 23 attributes collected and utilized in this research. Various approaches have been used in this research [9], the methods are described below:

1. Collection of data.
2. Data Preprocessing.
3. Chi-square Correlation.
4. Association Rule Mining.
5. Simulation Environment.

3.1 *Collection of Data*

We went to many psychiatrists to collect some questionnaires about depression. Then we collect the data from B.Sc., M.Sc., and Undergraduate students. Finally, we create our dataset after getting the answer. With this aim, a questionnaire has been designed to obtain information from students from special universities in Bangladesh for these findings. There are 539 instances in the dataset. For this study, we listed 23 attributes accumulated and used. The list of features appears in Tables 1 and 2.

3.2 *Data Preprocessing*

We have found out the missing data from each attribute [9]. We have cleaned the data through procedures such as filling missing values or removing missing data rows, smoothing noisy data, or resolving data inconsistencies. We have done a correlation to each feature particularly. After completing the correlation we got our most significant 18 features. Then we extracted our significant features. After extracting features, we used association rule mining techniques to get the dataset's most associated features. The whole system process design is shown in Fig. 1.

Table 1 Features (questions list) part 1

Features	Subcategory	Data distributions
		Percentage (%)
Age	20	9.10
	21	16.32
	22	15.02
	23	14.30
	24	14.84
	25	11.70
	Others	18.72
Gender	Male	66.60
	Female	33.40
Do you agree to all society rules?	Yes	66.05
	No	33.95
What do you think about you?	Extrovert	57.14
	Introvert	42.86
Do you have a smart phone?	Yes	93.13
	No	6.87
Do you like to spend your time with your friends?	Yes	89.23
	No	10.77
Are you happy with your current position?	Yes	66.04
	No	33.96
Are you tensed about your career?	Yes	71.24
	No	28.76%
Do you discuss your problems with anyone?	yes	61.03%
	No	38.97%
Do you feel comfortable being alone?	yes	61.23%
	No	38.77

3.3 Chi-Square Correlation

It is a concept that calculates the extent to which, to each other, two securities move. A chi-square test is essentially a data analysis of a random collection of variables based on observations. Typically it is a contrast between two sets of statistical data. It is also known as the χ^2 test. It also provides the specifics of the methods of statistical analysis that feature chi-square statistical selection methods for the aim. It helps the researcher to find the relations between the two variables. It is the compassionate movement of two or more variables. Firstly, for each of the functions, the chi-square χ^2 value is determined [10] (Fig. 2).

Table 2 Features (questions) list part 2

Features	Subcategory	Data distributions
		Percentage (%)
Did you try to committed suicide?	Yes	14.85
	No	85.15
What do you think that suicide is the solution to all problems in your life?	Yes	12.43
	No	87.57
Are you happy with your family members?	Yes	87.00
	No	13.00
Do you consider yourself as a family burden?	Yes	38.77
	No	61.23
Where you feel more comfortable?	Family	62.33
	Friends	29.87
	Others	7.80
How many members of your family?	4	36.73
	3	20.70
	5	21.33
	6	11.31
	7	3.71
	8	1.74
	Others	4.48
How much money does your family earn in a month?	above 40k	23.19
	30–40k	21.33
	30–25k	24.48
	20–25	17.25
	10–20	13.75
Have you participated extracurricular activities?	Yes	67.53
	No	32.47
Do you feel any difficulties in your educational system?	Yes	67.53
	No	33.17
How much time do you sleep in one day?	6h	18.93
	7h	27.82
	8h	25.99
	Above 8h	13.54
	Less than 6h	13.72

(continued)

Table 2 (continued)

Features	Subcategory	Data distributions
		Percentage (%)
What is your SSC result?	5	46.56
	4.5	7.42
	4	5.00
	4.88	4.26
	4.44	2.78
	4.56	31.53
	Others	2.45
What is your HSC result?	5	21.52
	4	12.43
	4.5	8.53
	3.5	3.90
	4.33	2.60
	4.75	2.60
	Others	48.42
What is your University CGPA?	3	12.80
	3.5	7.23
	2.5	3.14
	3.1	2.59
	3.25	2.22
	4	2.22
	Others	69.80

3.4 Association Rule Mining

The technique of association rule mining is intentional to identify recurrent trends, similarities, associations, or fundamental structures from data sets contained different forms of the database, for example, relational database, transactional database, and added data repository shapes [11]. We use the apriori algorithm in this study. It is an algorithm that attempts to manage database records, mainly transactional records, or records containing any number of fields or objects [12]. We can branch the algorithm into two parts. To get in a data set all the periodic groups with k number of the item. The periodic k-item set helps to use the rule of self-joining to find the periodic groups by item k+1. The entire process is illustrated in Fig. 3. It is an algorithm for the categorization apriori algorithm was used regarding the generator of frequent items. It also helps to correctly classify the association rules. Also, it is important to discover all the rules which have more help than threshold help, and greater confidence than the confidence level.

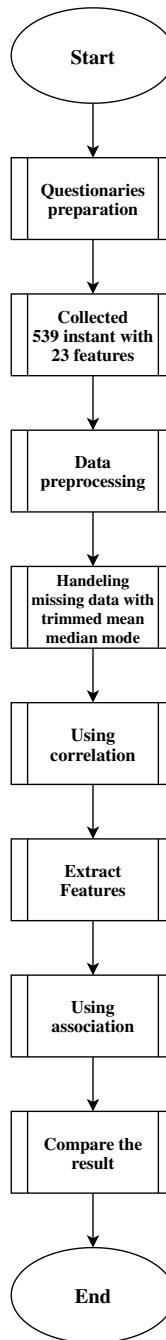


Fig. 1 System architecture of the study

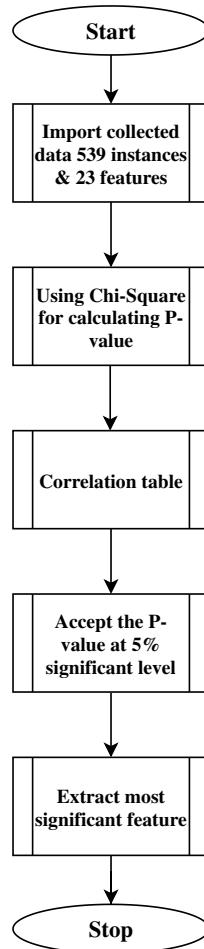


Fig. 2 Block diagram of correlation

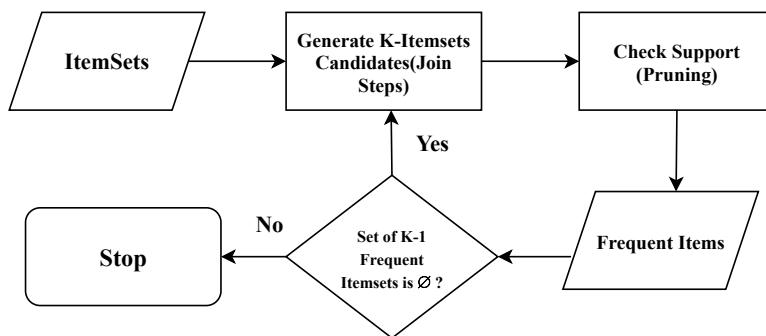


Fig. 3 Apriori algorithm process

3.5 *Simulation Environment*

- R-3.6.3
- R Studio 1.3.1093
- IBM SPSS 25.

3.6 *R Packages*

SPSS Statistics have been used for finding correlation. The R packages have been used for this research study. R is programmed to work the way it thinks about problems, R is both strong and versatile [13]. It is used widely for the study of both structured & unstructured data. It allows different features that distinguish it from other languages to measures in data science.

subset (): The functions of the subset return the vector, data from, etc. after meeting a specific condition.

chisq. test (): This package has used for the chi-square test.

apriori (): It lays down the rules of the association.

plot (): A common element for plotting.

inspect (): It is a representation of all options, plots, and statistics.

4 Outcomes

Now we showing the depression analysis table based on our data. The Depression analysis table has shown in Table 3.

We accepted the Asymptotic Significance of P-value 5% significant for feature selection. We have got these 18 significant features as P-Value accepts a 5% significant level. From the table, we have seen some features row have three columns (hopeless, satisfied, depression), which the Asymptotic Significance of P-Values are below 0.05 to each column. We have taken these features as highly significant depression stress. The Asymptotic Significance of P-Value is below 0.05 means it rejects the null hypothesis and satisfies the alternative hypothesis. Similarly, we have seen some features row have three columns (hopeless, satisfied, depression), which the Asymptotic Significance of P-Values are below 0.05 to two-column. We have taken these features as medium significant depression stress. And, we have seen some features row have three columns (hopeless, satisfied, depression), in which the Asymptotic

Table 3 P-Value of Different Features vs Outcome

Features	Hopeless (P-value)	Satisfied (P-value)	Depression (P-value)
Age	0.006	0.778	0.042
Gender	0.001	0.743	0.654
Do you agree to all society rules?	0.000	0.151	0.767
What do you think about you?	0.064	0.163	0.055
Do you have a smart phone?	0.0001	0.103	0.583
Do you like to spend your time with your friends?	0.0001	0.024	0.004
Are you happy with your current position?	0.001	0.001	0.001
Are you tensed about your career?	0.145	0.0001	0.0001
Do you discuss your problems with anyone?	0.0001	0.001	0.0001
Do you feel comfortable being alone?	0.152	0.597	0.067
Did you try to committed suicide?	0.0001	0.0001	0.0001
What do you think that suicide is the solution to all problems in your life?	0.0001	0.0001	0.0001
Are you happy with your family members?	0.0001	0.0001	0.0001
Do you consider yourself as a family burden?	0.000	0.0001	0.0001
Where you feel more comfortable?	0.0001	0.0001	0.0001
How many members of your family?	0.293	0.298	0.087
How much money does your family earn in a month?	0.239	0.947	0.671
Have you participated extracurricular activities?	0.001	0.0001	0.0001
Do you feel any difficulties in your educational system?	0.025	0.164	0.0001
How much time do you sleep in one day?	0.125	0.283	0.0001
What is your SSC result?	0.0001	0.373	0.016
What is your HSC result?	0.009	0.226	0.161
What is your University CGPA?	0.348	0.719	0.825

Significance of P-Values are below 0.05 to one column. We have taken these features as low significant depression stress.

Here, We have got 9 features as highly significant depression stress. These 9 features are given below:

- (i) Do you like to spend your time with your friends? (Like_spend_Time_with_Friend)
- (ii) Are you happy with your current position? (Happy_Current_position)
- (iii) Do you discuss your problems with anyone? (Discuse_Problem_Anyone)
- (iv) What do you think that suicide is the solution to all problems in your life? (Suicide_Is_The_Solution_all_problems)
- (v) Did you try to commit suicide? (Try_Committed_Suiside)
- (vi) Are you happy with your family members? (Happy_With_Family_Members)
- (vii) Do you consider yourself as a family burden? (Consider_yourself_Family_burden)
- (viii) Where you feel more comfortable? (Where_Feel_More_Comfortable)
- (ix) Have you participated extracurricular activities? (Participated_extra_curricular_Activities).

Also, we have got 4 features as Medium significant depression stress. These 4 features are given below:

- (i) Age
- (ii) Are you tensed about your career? (Tensed_About_Carrer)
- (iii) Do you feel any difficulties in your educational system? (Difficulties_in_your_educational_system)
- (iv) What is your SSC result? (SSC_Result).

And we have got 5 features as lower significant depression stress. These 5 features are given below:

- (i) Gender
- (ii) Do you agree with all society's rules? (Agree_society_rule)
- (iii) Do you have a smartphone? (Have_Smart_Phone)
- (iv) How much time do you sleep in one day? (Sleep_in_a_day)
- (v) What is your HSC result? (HSC_Result).

Association rules are completed via scanning information for successive if-then forms. Confidence indicates the occasions the if-then statements are discovered true. A third measurement, called lift, can be used to separate confidence with estimated confidence. Apriori algorithm have been used for Association Rule Mining. The outcomes have been shown in Table 4. Total of 8 rules has been obtained where the Support = 45% and Confidence = 95%.

Figure 4 introduces the scatter plots and the graphs showing a dataset relationship between two variables. It stands for data points on a two-dimensional plane or the Cartesian system. When you move from left to right, decode a scatter plot by penetrating into the data for patterns. This describes an uphill relation between X

Table 4 Association Rules with Support (S), Confident (C) and Lift (L)

SL no.	LHS	RHS	S	C	L
1	Have_Smart_Phone=yes, Like_Spend_Time_With_friend=yes, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.549	0.980	1.126
2	Have_Smart_Phone=yes, Like_Spend_Time_With_friend=yes, suicide_is_the_solution_all_problems=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.5194	0.979	1.125
3	Have_Smart_Phone=yes, Like_Spend_Time_With_friend=yes, Try_Committed_Suicide=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.508	0.978	1.124
4	Have_Smart_Phone=yes, Like_Spend_Time_With_friend=yes, Try_Committed_Suicide=no, suicide_is_the_solution_all_problems=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.489	0.9777	1.123
5	Like_Spend_Time_With_friend=yes, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.567	0.977	1.123
6	Have_Smart_Phone=yes, Try_Committed_Suicide=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.532	0.976	1.121
7	Like_Spend_Time_With_friend=yes, suicide_is_the_solution_all_problems=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.530	0.976	1.121
8	Like_Spend_Time_With_friend=yes, Try_Committed_Suicide=no, Where_feel_more_comfortable=family	happy_with_family_members=yes	0.525	0.975	1.121

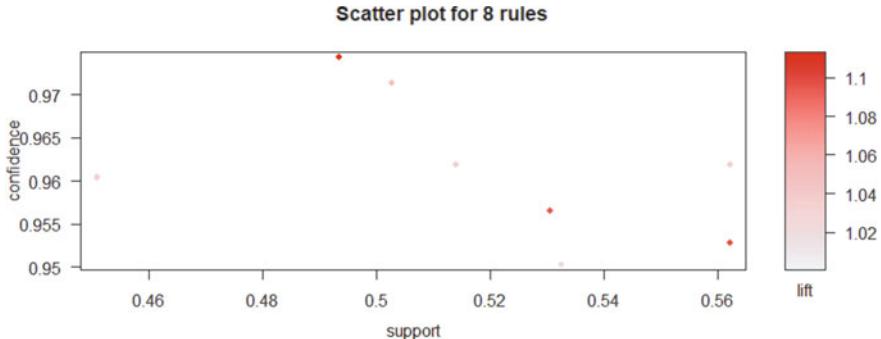


Fig. 4 Scatter plot for 8 rules

and Y if the data determines a positive trend when you go from left to right. The Y-values would usually increase as the X-values increase (move right) (move up). Graph-based representation is especially appropriate when the examiner is keen on an aggregated perspective on the most significant rules. Graph-based visualization procedures provide an incredibly simple representation of rules for moderately limited arrangements of the most relevant rules with respect to association rule mining, which can be reliably selected depending on their corresponding lift ratings.

Figure 5 introduces the graph visualization for the most significant extracted association rules. In the system chart, item sets are introduced as vertices though rules are introduced as coordinated edges between item sets. Each factor is given its axis in a Parallel Coordinates Plot and all the axes are set parallel to each other. And Fig. 6 also introduces all axes that can contain an alternate scale, as each factor works off an alternate unit of evaluation, or all the axes can be consistent to remain all the scales uniform. The values are plotted as a sequence of lines associated with all the axes. This illustrates that every line is a group of points positioned on each axis, that have all been associated jointly.

Fig. 5 Group for 8 rules

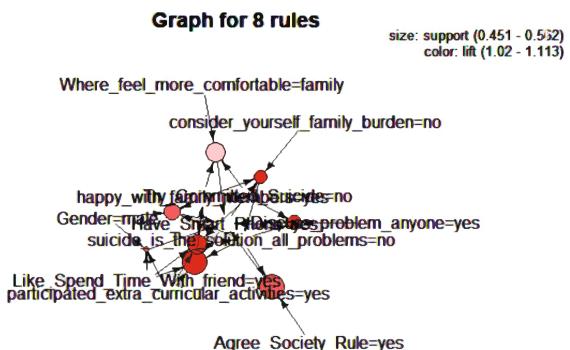
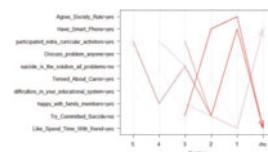


Fig. 6 Parallel coordinates plot for 8 rules



5 Conclusion

This research was established by the student outcome factors using the data collected from several universities. The side effects of depression can also be seen at work, at school, at home. It is a matter of sorrow that the depression rate is increasing day by day. Information from students had been collected via the survey, and there had many missing data in those from the data set. In this analysis, total 9 features were shown highly significant, 4 features as medium, and 5 features as lower significant. Total 8 significant rules were found in this research study. Apriori algorithm was applied to find significant rule. For rule mining, the support and confidence was 0.45 and 0.95 respectively. In the study, several limitations were faced. The number of instances was not sufficient. As this research was conducted during this pandemic situation, on the field level data collections were not possible. In Future, classification of the depression level will be conducted using different algorithms. For instance, Artificial Neural Network, Random Forest, different Boosting Technique, and Fuzzification will be used to classify the stress level of the human.

Acknowledgements We are deeply thankful to our parents. They have always been inspired and motivated to achieve our aims. Also, this research would not be made possible without the knowledge and assistance of students from several universities. We would like to thanks Dr. Mahabubo Kibria (MBBS, MD), professor of Khulna Medical College, Khulna, Dr. Tasnim Sahbah (MBBS), City Medical College, Khulna, Dr. Sharmin Akhter (MBBS, MD), Bangabandhu Sheikh Mujib Medical University, Dr.Istiyak Ahmed (MBBS), Gazi Medical College, Khulna, Dr. Najninkhan Nipa (MBBS), Ragib Rabeya Medical College, Sylhet and Dr. Kamal Uddin (MBBS), KUET for their valuable comments and suggestions regarding the attributes (symptoms) used in this study.

References

1. F. Li, Y. Ding, Data mining in Cognitive function training of depression patients applications, in *10th International Conference on Information Technology in Medicine and Education (ITME)* (2019), pp. 98-101
 2. Depression, *Who.int* (2020), <https://www.who.int/news-room/fact-sheets/detail/depression>. Accessed: 01 Aug 2020
 3. Gender differences in suicide (2020), *En.wikipedia.org*. https://en.wikipedia.org/wiki/Gender_differences_in_suicide. Accessed 10 Aug 2020
 4. D. Ramalingam, V. Sharma, V. Sharma, Study of depression analysis using machine learning techniques. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)* **8**(72) (2019)

5. A. Sharma, W. Verbeke, Improving diagnosis of depression with XGBOOST machine learning model and a large biomarkers dutch dataset ($n = 11,081$). *Front. Big Data* **3**, 1–11 (2020). <https://doi.org/10.3389/fdata.2020.00015>
6. M. Deshpande, V. Rao, Depression detection using emotion artificial intelligence, in *International Conference on Intelligent Sustainable Systems (ICISS)* (Palladam, 2017), pp. 858–862. <https://doi.org/10.1109/ISS1.2017.8389299>
7. F. Cacheda, D. Fernandez, F. Novoa, V. Carneiro, Early detection of depression: social network analysis and random forest techniques. *J. Med. Internet Res.* **21**(6), e12554 (2019). <https://doi.org/10.2196/12554>
8. K. Shrestha, Machine learning for depression diagnosis using twitter data. *Int. J. Comput. Eng. Res. Trends* **5**(2), 56–61 (2018)
9. M. Raihan, M. Islam, P. Ghosh, J. Angon, M. Hassan, F. Farzana, A machine learning approach to identify the correlation and association among the students' educational behavior, in *Proceedings of the International Conference on Computing Advancements* (New York, USA, 2020), pp. 1–6. <https://doi.org/10.1145/3377049.3377130>
10. Chi-Square Test (Definition, Formula, Properties, Table & Example), *BYJUS* (2020), <https://byjus.com/math/chi-square-test/>. Accessed 02 Sep 2020
11. What is the apriori algorithm?, *Techopedia* (2020), <https://www.techopedia.com/denition/33156/apriori-algorithm>. Accessed 05 Sep 2020
12. Why use the R Language?, *Burns Statistics* (2020), <https://www.burns-stat.com/documents/tutorials/why-use-the-r-language/>. Accessed 10 Sep 2020
13. Why learn R? 10 handy reasons to learn R programming language, *DataFlair* (2020), <https://data-air.training/blogs/why-learn-r/>. Accessed 15 Sep 2020

Implementation of A Smart Helmet with Alcohol and Fall Detection and Navigation System



Piyush Mishra, Pratik Pai, Pradhuman Singh, Vedant Kayande, and Manish Parmar

Abstract Since the invention of the first motor vehicle, the automobile industry has been advancing by leaps and bounds. Two-wheelers are no exception, but the safety of the driver is still a precarious situation. Due to the very built and design of the two-wheeler, the rider is more vulnerable to accidents as compared to the traditional four-wheeler vehicle. Therefore, in this paper we propose a holistic solution for the safety of the driver of a two-wheeler. The proposed solution aims to perform three main tasks. Firstly, alcohol detection which would be required to detect whether the rider is under the influence of alcohol. Secondly, a fall detection system which would be required to send notifications to the emergency contacts in case of an accident or a serious fall from the two-wheeler. Lastly, a navigation system to enhance the user experience. Our solution consists of a synchronous working of both hardware and software. For both the alcohol and fall detection we would use sensors which would feed the data to the microcontroller, which in turn would compare the data with the threshold values in real time. By pushing notifications to the relevant emergency contacts, we have managed to make use of the Internet of Things which always helps the system to remain connected to the Internet. The details of the implementation have been elaborated in this paper. In addition, the implementation and observations have also been highlighted to solidify our proposed solution. We believe such a system

P. Mishra (✉) · P. Pai · P. Singh · V. Kayande · M. Parmar

Department of Electronics and Telecommunication Engineering, Sardar Patel Institute of Technology, Andheri West, Mumbai 400058, India

e-mail: piyush.mishra@spit.ac.in

P. Pai

e-mail: pratik.pai@spit.ac.in

P. Singh

e-mail: pradhuman.singh@spit.ac.in

V. Kayande

e-mail: vedant.kayande@spit.ac.in

M. Parmar

e-mail: manish_parmar@spit.ac.in

can not only help secure the life of the rider of the two-wheeler but also improve the experience one has while driving.

Keywords Smart helmet · Fall detection · Alcohol detection · Navigation system · Internet of Things · Microcontroller

1 Introduction

As mentioned above, the proliferation of technology has enabled us to witness massive advancements in pretty much every sector one can think of. The automobile sector has been an industry which has embraced technology with open arms. It has constantly been including technology not only to enhance the user experience but also to make the experience much less cumbersome. It is not uncommon, in today's world, to see an automobile packed with cutting edge technology. However, the two-wheeler sector has not been exploited as much as the four-wheeler sector. An overall analysis of the accidents with proper segmentation on basis of vehicle has been highlighted in [1] which gives valuable data insights. Given the fact that the rider of the two-wheeler is much more vulnerable to accidents, we believe there is a much-needed improvement in the safety of the riders. There has been an alarming increase in the number of road accidents in India and particularly in the accidents of two-wheeler. The statistics show that 110,777 people die every year because of motorcycle accidents. Every year, about 300,000 teenagers go to the emergency department because of bike injuries. This number has almost doubled since 1997. This alarming increase has caught attention of numerous governments. Further, these accidents can be directly attributed to certain factors such as driving under the influence of alcohol, driving at very high speeds and most importantly, data showed that if the drivers involved in the road accidents were attended or were taken to the hospital much quicker, then a huge number of deaths could be avoided. In many cases the injury suffered by the rider is a minor one, but due to the late arrival of paramedics, sometimes these very minor injuries can turn out to be fatal. We have tried to address these issues in our proposed solution and have tried to provide a holistic solution. The entire hardware required to implement our solution can easily be mounted on any helmet and hence our solution can be used ubiquitously due to its all-compatible nature. This can prove to be a major advantage and hence can justify its use by all the riders. The hardware used consists of extremely compact sensors which serve various purposes and the software used is primarily implemented via the smartphone of the user. We have proposed to alert the specified users in case of an emergency. Emergency can be broadly defined as the following cases: 1. Fall Detected in case of an accident 2. When the rider is driving under the influence of alcohol.

In any of the above instances, the sensors help detect these cases and a push notification in the form of text message is provided to the relevant contact. Lastly, we have also incorporated a navigation system which makes best use of the google maps. A mic is used to activate the Google assistant.

The paper consists of all the above-mentioned features which have been implemented. Firstly, the research about current and existing methods was carried out in order to make the most of the implementation. This has been explored in the Literature survey of the paper. Post that, the detailed methodology and algorithm deployed has been explained in detail. Both have been arrived after reviewing all possible solutions and using our own innovation. This is followed by the result and analysis of the implementation which is basically the proof for our work and provides strong backing to what was proposed initially in the paper and what has been achieved. In this paper, the proposed solution has been implemented in its maximum capacity. Finally, in the conclusion, the future scope of the paper has also been discussed to show further research possible.

2 Literature Survey

The usage of hardware and software for creation and design of many products is not a novel concept. They both have been used optimally to produce the desired result. The basic idea we have implemented in this project lies in the very primitive form of the most common sensor integration techniques. Sensors are used to detect and measure certain parameters, which are then utilized by the software program which decides how the data would be handled. This data is then converted into meaningful data and the processor performs the relevant functions with respect to input parameters which are essentially the sensor data. In this paper, we have combined various sensors and acquired the data correspondingly to produce commensurate results. We require the several sensors for various data points which would enable us to provide the holistic solution which we have proposed. Firstly, for the implementation of the alcohol detection system, we have planned to use a single sensor module, which would be mounted smartly in order to take the most accurate readings. The sensor used would take in the breath of the user as the input [2]. This would then be fed into our algorithm to check for the threshold values. Depending upon the threshold values the required action is taken. If the threshold value is exceeded, then the notification is sent to the emergency contacts. The threshold value has been taken into consideration by analyzing [3] and [4]. The conversion of the blood alcohol level will be explained in detail in the later parts of this paper. Secondly, for the implementation of the fall detection system, we have used an accelerometer which provides us with a 6-axis range of motion [5]. This helps us increase our accuracy of detection and combat the natural head movements which the driver might have. The basic idea was inspired by [6] and the accuracy of the detection has been improved. Coupled with this, this paper incorporates the push notification system, which alerts the emergency contacts in case of the detection of fall. Since we have taken into considerations various possibilities (which is further explained in the paper) the detection accuracy is high. Lastly, to integrate all of them and to send the push notifications we have used a Global Positioning System (GPS) module. In any of the above-mentioned cases, if an emergency is detected, the emergency contacts are alerted. The alert consists

of a text message and the GPS location of the driver. This would aid the concerned authority to take prompt action and, in many cases, might prevent fatality. In [7] detail about the above implementation is stated and we have adopted a similar version and have excluded the Global Systems for Mobile Communication (GSM) module, to lower the costs.

Given the advancements in technology a need for the holistic solution, we have implemented all the above things into a single product to leverage all the functionality of the individual sensors and enhance the user experience. Every module might yield highly accurate results but since accidents can be attributed to a myriad of factors, the incorporation of all the factors is a much-needed advancement.

3 Methodology

As mentioned in the above part, our project can be broadly classified into the following parts:

1. A Fall Detection system for accident alerts
2. Alcohol Detection System to prevent drunk driving
3. Navigation System.

Combining these three would provide the holistic solution which would help realize the paper's objective of a Smart Helmet. These objectives have been explained individually in the methodology and a combined overview of the algorithm deployed is stated in the further parts of this paper.

3.1 Fall Detection System

As mentioned in the above parts the main objective of the Fall Detection system is to detect an accident of a rider of a two-wheeler vehicle. When one considers the rider of a two-wheeler vehicle and the position of the helmet worn by him/her, it is evident that the helmet is susceptible to various forms of disturbances and changes in orientation. For example, when a rider is wearing the helmet, he/she would pick it up from the storage area and take it to the two-wheeler vehicle for the future use. For implementation such extreme conditions are excluded. Coupled with this, the conditions when the rider takes the helmet and has a brisk walk with it to any destination are also excluded for the same purpose. Such extreme conditions could be taken care of by various methods which have been stated in the Future scope.

However, there are various cases which can pose a difficult scenario for the implementation. Since the sensors are deployed on the helmet, there would still be a fair amount of oscillations, disturbances, or changes in axes which should be considered to have higher accuracy. For example, when the rider is riding the two-wheeler vehicle, he/she would change the orientation of their neck/head either due to fatigue,

itching, or any other reasons. This would affect the readings of the sensor. Further, such changes are very prevalent and occur very often while riding a two-wheeler vehicle. Purushothaman et al. [8] implement a similar anomaly which occurs in senior citizen's fall. Artificial Neural Networks (ANN) helps include the outlying cases and provides a more accurate solution for the wide range of movements which occur in senior citizens. A similar implementation is done in [9] but backpropagation has been used in this case. The threshold for the fall detection has been decided to keep the above factors in mind. The derivation of the same is shared in the algorithm. For the sensor, the paper implements the fall detection using the MPU6050. The reason for choosing the MPU6050 is due to its patent advantages over the other sensors. It provides a 6-axis accelerometer which helps us gain higher accuracy for the fall detection. Further, it being a compact device with a small number of ports it helps provide comfort to the users by not involving a large amount of connections which would increase the cumbersome nature of the helmet. Since MPU6050 also comes with a gyroscope, one does not require extra hardware to achieve the objective. Therefore, the paper has implemented fall detection using MPU6050.

3.2 Alcohol Detection System

The second key aspect focused in the paper is the detection of alcohol in the breath of the rider of the two-wheeler vehicle. Given that the alcohol content in the breath of a person has been analyzed for a long period of time post the invention of the breath analyzer, the implementation in the paper has also drawn inspiration from the same.

The novelty of the paper is further enhanced by the positioning of the sensor is unique. It has been strategically placed on the helmet close to the mouth of the rider so that the rider can conveniently breath into the center and the required results can be obtained. In addition, the placement of the sensor has also considered the interference between the head of the rider and the helmet when the rider is putting on the helmet or taking off the helmet. This provides a great advantage as it prevents and protects the sensor.

The sensor used is MQ3 gas sensor. The reception quality of the sensor and the usage of this sensor in the world-renowned breath analyzer. Tapadar et al. [10] explain the inherent advantages of the MQ3 sensor in the alcohol detection.

3.3 Navigation System

Given the proliferation of navigation systems throughout the world, it has been easier for us humans to reach from one place to the another. Such systems are more helpful in conditions when the commuter does not know the location of the destination. Plugging in the name of the location reveals an ocean of information about the location and devises a driving route toward the location. In recent times, these systems

have also taken into consideration the traffic conditions which further ameliorate the experience. However, one big shortcoming of the system is that one needs to operate one's mobile device or similar devices to make use of the navigation system. While in a four-wheeler, given the smart technologies, such a system can be used easily. But, in the case of the two-wheeler one needs to always stop and look at one's mobile device in order to make use of the navigation system. Further, incorporating a GPS module for real-time tracking and a module to communicate with the Internet would be extremely cumbersome and difficult to implement as shown in [11], which has only explored tracking via the GPS location. If it were to include navigation, then the system would be ten times more complicated.

This paper provides an elegant solution for the above problem by incorporating the google voice assistant and google maps. A single piece of hardware, which is the mic is included in the helmet to facilitate the same. The microphone used is shown in the adjoining figure and it ensures the highest quality of sound grasping in order to recognize and analyze the speech input.

3.4 Microcontroller and Implementation

For the integration of all the components as discussed above, the seamless implementation of the same would require a microcontroller. The microcontroller would act as the main functional unit which controls and regulates the action of all the other components mentioned above. Since the implementation would require Internet connectivity and the implementation would require us to send a lot of data over the Internet, the microcontroller chosen is ESP8266. This provides a strong Internet connectivity and serves the purpose. The paper [12] further bolsters the point and provides valid points in favor of esp8266.

Further, the biggest advantage of the esp8266 is its compact size. Since, the paper has stressed upon the implementation of smart helmet, which must be comfortable, the compact nature of esp8266 further reinforces the same fact. Lastly, the architecture also provides a high level of durability and a strong power saving architecture. Due to these patent advantages, the esp8266 justifies its inclusion for the microcontroller.

For the implementation purpose, a web application has been developed which portrays an interactive dashboard. All the readings from the sensor are fed to the microcontroller which in turn is connected to the Internet as mentioned above. This makes the transfer of data seamless and allows a large amount of data to be sent. The Web app offers real-time measurement of both the alcohol and the readings from the MPU6050. This dashboard can be seen for reference of different values and can be used to base models which could enhance the efficiency of the model. Further, as soon as the threshold of any of the set parameters is broken an alert notification is sent via the Internet to the concerned emergency contacts. Since everything is done via the Internet the speed of implementation is significantly enhanced.

4 Algorithm

The flow of the algorithm is dependent upon the three divisions made above. All the three factors which are considered, i.e., Alcohol Detection, Fall Detection, and Navigation system would be required to be working all the time and in tandem. The algorithm devised in this paper aims to serve that purpose and has taken into consideration the working of all the three divisions. Figure 2 shows the detailed flow and working of the algorithm. It gives a broad overview of the entire algorithm. The working of the algorithm is mentioned below.

Initially, implementation of the alcohol detection system is important as it serves as the very first step. The alcohol is detected when the user/rider of the vehicle breathes air out of his mouth into the sensor when the rider wants to start the journey. The air from his mouth enters the sensor and depending upon the values returned by the sensor required action is taken. The values of the threshold have been calculated based on the calculations as shown in Equation. 1 and the graph in Fig. 3. The graphs give very deep insight and relation between the sensor values and the accepted alcohol levels in the human body which is also referred to as Blood Alcohol Content (BAC). After determining the threshold, that value is stored in the memory and is used to compare to all the input values which follow. The input value is fetched from the MQ3 sensor and fed to the microcontroller for comparison. If this value is lesser than the threshold value, then the rider is safe to ride the two-wheeler. If it is above the threshold value, then ideally the rider should not drive the two-wheeler and therefore, the emergency contacts should be contacted in this case. A protocol is sent to the microcontroller which via the nodeMCU connects to the Internet and a text

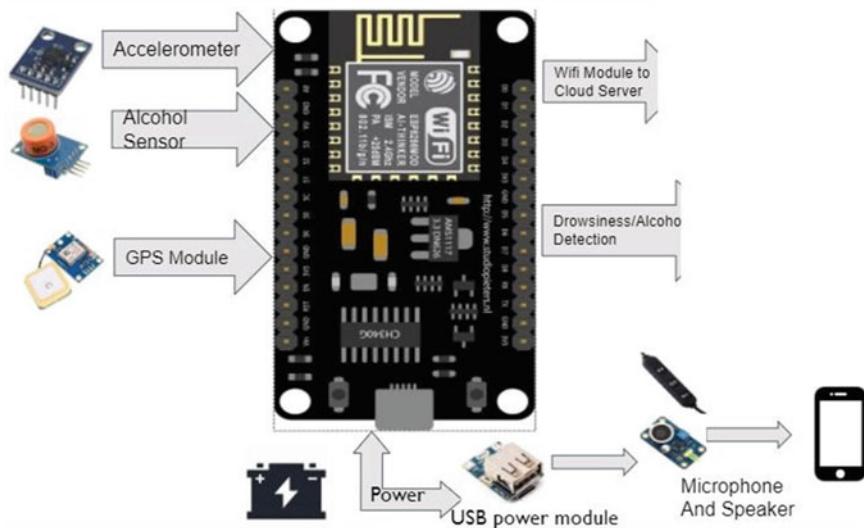


Fig. 1 Block diagram of the entire implementation

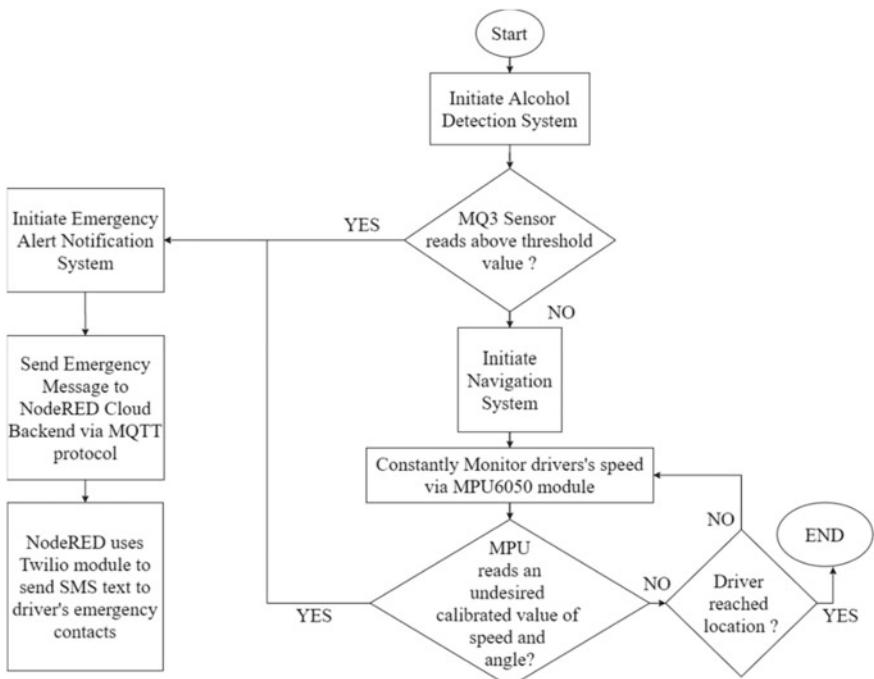
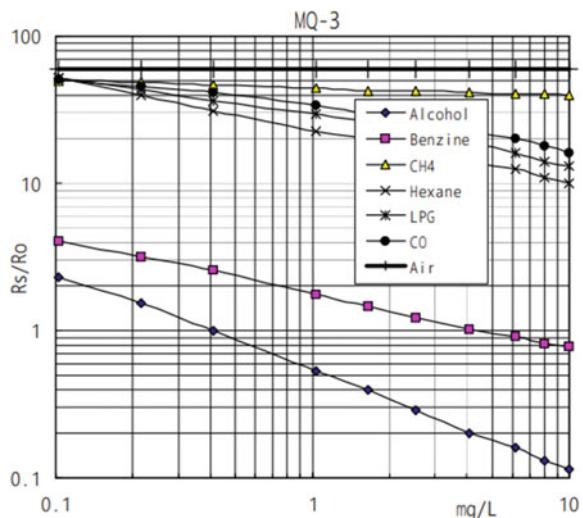


Fig. 2 Flowchart of the entire implementation

Fig. 3 Sensitivity characteristics of the MQ3 sensor



message is sent to the emergency contact. Such a flow will be observed throughout the implementation for sending the alert text messages to the emergency contacts.

Having done the initial step of detecting the alcohol content in the rider, if permissible, the rider rides the bike safely. After this, the second part is the navigation system. This follows a simple algorithm because it makes use of several inbuilt features in the mobile device which is owned by the user. Firstly, the mic is fitted to the helmet as stated above. Whenever the user wants to access the mic, he/she may do so by speaking into it and initiating whatever protocol that he/she may wish to initiate. This speech is then transferred directly into the smartphone of the user and the voice assistant from Google is activated. Post this, the user can easily command whatever navigation help he/she requires.

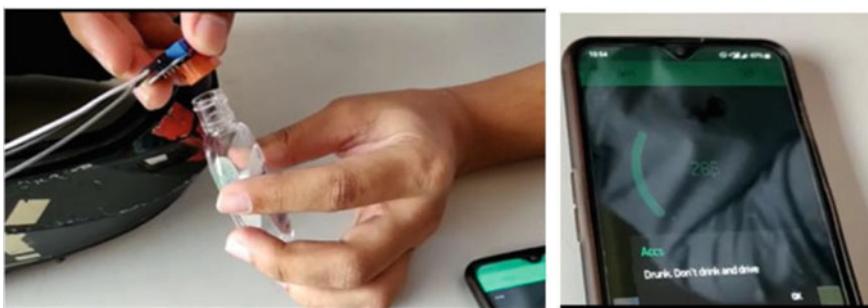


Fig. 4 Detection of alcohol and alert message (for implementation purposes Sanitizer has been used since it has alcohol)

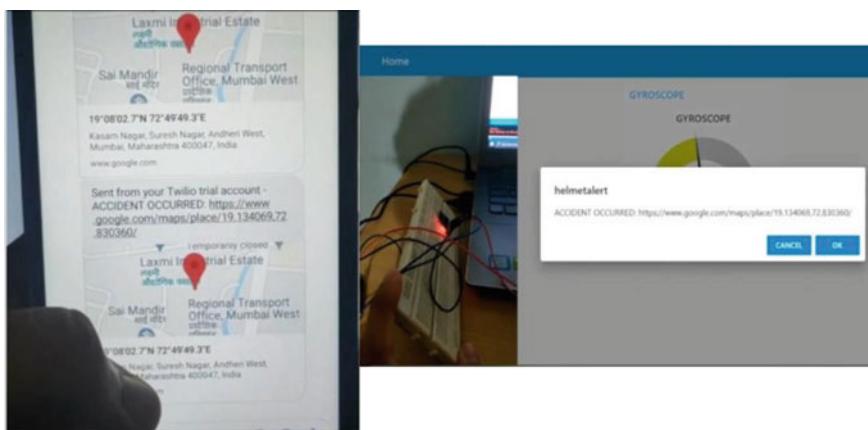


Fig. 5 Fall Detection as seen by moving of the gyroscope past the threshold and notification as a text message with the live location

Lastly, as shown in the flow chart, the fall detection system is to be implemented. This system is implemented using MPU6050 which has a gyroscope and a accelerometer. The yaw, pitch, and roll angles are continuously monitored using a feedback loop and checked if they are within their threshold values. These angles are calculated using the values from gyroscope (96%) and accelerometer (4%). Offset calibration is required to get accurate values. There is an issue of drift in gyro values which is corrected using the accelerometer values. After the value retrieved from the sensor crosses the defined threshold value, then the fall is said to be detected. Post this, the same protocol is followed of alerting the emergency contacts and make them cognizant of the fall. The location and the time of the fall both are mentioned in the alert message for the emergency contact to act accordingly.

$$Ro = 0.050698 \text{ at fresh air stability} \quad \text{sensor_volt} = \frac{\text{sensor value}}{1024} * 5$$

$$Rs = (5 - \text{sensor_volt}) / \text{sensor_volt}$$

$$Ro = \frac{Rs}{60} \text{ (60 is found via interpolation)}$$

$$\text{For } 30 \frac{\text{mg}}{\text{L}} \rightarrow \frac{Rs}{Ro} = 14.8322$$

Sensor Volt = 1.5382V Sensor Value = 313.3

Calculation for the conversion to voltage value:

$$\frac{\text{mg}}{\text{L}} = 0.189563503 \left(\frac{Rs}{Ro} \right)^2 - 8.6177665431 - \frac{Rs}{Ro} + 1.079213151$$

Equation 1. The calculations involved in determining the Blood Alcohol Content

5 Analysis and Results

The ESP8266 is initialized and code is uploaded using Arduino. For fall detection, we have used MPU6050 which gives us 6-axis readings. In order to remove false negatives, we have set up a threshold angle. Whenever helmet crosses this threshold angle, a notification alert is seen on the web application. In the pictures, we can see when the helmet is in stationary position, gyroscope readings are constant. Whereas when the helmet is tilted, the gyroscope readings start to change. As we can see in the pictures, when the angle crosses the threshold, web applications give us the notification. For alcohol detection, we have used MQ3 sensor. MQ3 sensor measures the alcohol level very accurately. Whenever the sensor reaches a threshold mark, a notification alert is seen on the web application. In the pictures, we can see when the alcohol is not detected, MQ3 sensor readings are constant. Whereas when the alcohol is detected, the sensor readings start to increase. When the level crosses the

threshold, web application gives us the notification. For GPS, we have used NEO-6M sensor. Whenever a fall is detected, the GPS sensor gives us the latitude and longitude co-ordinates of the vehicle. Web application plots the co-ordinates and gives us a very precise location of the driver.

6 Conclusion

From the results and analysis, we can conclude that the implementation of the smart helmet is feasible using a set of sensors and a microcontroller and stable Internet connection. The accuracy of the implementation is something which cannot be quantified at the present stage. However, certain extrapolation about the accuracy of the model can be made using the threshold values and how accurate these values are. For the threshold values of both the Blood Alcohol Content (BAC) and for the fall detection have been decided after much experimentation. However, there would remain scope for improvement.

Depending upon the requirement the specific part of the implementation is triggered. As it can be seen in the analysis and results sections, the output for the triggered part will remain the same and would mostly be in those realms. Given that the microcontroller used is nodeMCU, it can only accommodate a couple of sensors and in case of more sensors a more powerful microcontroller can be used to both improve the power usage and the efficiency of the system. The web application so formed served a purpose of a strong user interface and can be used to get further crucial data insights.

Given certain areas where the paper can be improved upon further research, the future scope of the paper holds the following prospects:

Firstly, an important cause of accidents, particularly in India has turned out to be drowsiness. Drowsiness can occur due to several reasons and it leads to severe lapses in concentration. As driving, especially that of a two-wheeler vehicle, always requires the driver to be attentive and react to changes in the environment in the quickest and the most efficient way possible, drowsiness can cause several accidents. A driver feeling sleepy should either take a break for some time or avoid driving until he/she feels they are ready for the daunting task. Including a drowsiness detection system in the solution explained in the paper would provide an even more holistic solution as it would take into consideration almost all the factors which may lead to accident, therefore assuring maximum safety of the rider at any times. Mesquita et al. [13] and Khunpisuth et al. [14] show the implementation of the same but have a huge amount of hardware, which can be reduced.

Secondly, the cost of the solution proposed in this paper can further be reduced if all the sensors and the microcontroller are combined in a singular system. This can help prevent cost and given the target demographics of the users, it will further facilitate the sales. Coupled with this, whenever any product is produced at scale the price of production/manufacturing of the same product reduces dramatically. If it is

produced really at a very high volume, then the prices of the traditional helmet would be comparable to the price of the smart helmet.

Lastly, several comfort features can be added to enhance the user experience of the product and provide the most comfortable and conducive environment for the rider of the two-wheeler vehicle. This would significantly enhance the relevancy of the smart helmet in the current market and would successfully dethrone the current major players.

References

1. S.K. Singh, Road traffic accidents in India: issues and challenges. *Transp. Res. Procedia* **25**, 4708–4719 (2017). ISSN 2352-1465
2. S. Al-Youif, M.A.M. Ali, M.N. Mohammed, Alcohol detection for car locking system, in *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (Penang, 2018), pp. 230–233. <https://doi.org/10.1109/ISCAIE.2018.8405475>
3. The effects of a change in permissible blood alcohol concentration limit on involving drink-driving in road accidents. EmirSmailović, DaliborPešić,NenadMarković,BorisAntić,KrstoLipovac Faculty of Transport and Traffic Engineering, Vojvode Stepe 305, Belgrade 11000, Serbia
4. K. Sandeep, P. Ravikumar, S. Ranjith, Novel drunken driving detection and prevention models using Internet of Things, in *2017 International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT)* (Warangal, 2017), pp. 145–149. <https://doi.org/10.1109/ICRTEECT.2017.38>
5. A.Z. Rakhman, L.E. Nugroho, W. Kurnianingsih, Fall detection system using accelerometer and gyroscope based on smartphone, in *2014 The 1st International Conference on Information Technology, Computer, and Electrical Engineering* (Semarang, 2014), pp. 99–104. <https://doi.org/10.1109/ICITACEE.2014.7065722>
6. N. Noury, P. Rumeau, A.K. Bourke, G.O. Laighin, I.E. Lundy, A proposal for the classification and evaluation of fall detectors. *IRBM* **29**(6), 340–349 (2008)
7. U. Bharavi, R.M. Sukesh, Design and development of GSM and GPS tracking module, in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)* (Bangalore, 2017), pp. 283–288. <https://doi.org/10.1109/RTEICT.2017.8256602>
8. A. Purushothaman, K.V. Vineetha, D.G. Kurup, Fall detection system using artificial neural network, in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* (Coimbatore, 2018), pp. 1146–1149. <https://doi.org/10.1109/ICICCT.2018.8473219>
9. A. Jefiza, E. Pramunanto, H. Boedinoegroho, M.H. Purnomo, Fall detection based on accelerometer and gyroscope using back propagation, in *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)* (Yogyakarta, 2017), pp. 1–6. <https://doi.org/10.1109/EECSI.2017.8239149>
10. S. Tapadar, S. Ray, H.N. Saha, A.K. Saha, R. Karlose, Accident and alcohol detection in bluetooth enabled smart helmets for motorbikes, in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (Las Vegas, NV, 2018), pp. 584–590. <https://doi.org/10.1109/CCWC.2018.8301639>
11. J.M.M. Khin, N.N. Oo, Real-time vehicle tracking system using Arduino, GPS, GSM and web-based technologies. *Int. J. Sci. Eng. Appl.* **7**(11), 433–436 (2018). ISSN: 2319-7560
12. J. Mesquita, D. Guimarães, C. Pereira, F. Santos, L. Almeida, Assessing the ESP8266 WiFi module for the Internet of Things, in *2018 IEEE 23rd International Conference on Emerging*

- Technologies and Factory Automation (ETFA)* (Turin, 2018), pp. 784–791. <https://doi.org/10.1109/ETFA.2018.8502562>
- 13. O. Khunpisuth, T. Chotchinasri, V. Koschakosai, N. Hnoohom, Driver drowsiness detection using eye-closeness detection, in *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)* (Naples, 2016), pp. 661–668. <https://doi.org/10.1109/SITIS.2016.110>
 - 14. K. Satish, A. Lalitesh, K. Bhargavi, M.S. Prem, T. Anjali, Driver drowsiness detection, in *2020 International Conference on Communication and Signal Processing (ICCP)* (Chennai, India, 2020), pp. 0380–0384. <https://doi.org/10.1109/ICCP48568.2020.9182237>

BlockFITS: A Federated Data Augmentation Modelling for Blockchain-Based IoVT Systems



Bhrigu Kansra , Harshita Diddee , Tariq Hussain Sheikh, Ashish Khanna, Deepak Gupta , and Joel J. P. C. Rodrigues

Abstract In Intelligent Transport Systems (ITS), the collection of diverse data is a major practical roadblock; not only can their data be personally identifiable, i.e. private, but also the lack of incentive for entities to participate in any kind of collaborative training is also severely limited due to the added computational expense of training collaborative models locally. In this paper, we propose BlockFITS: A Vehicle-to-BlockChain-to-Vehicle (V2B2V) federated learning enabled model training paradigm for ITS entities. In addition to which we propose a data augmentation scheme that operates with cooperative training to generate an incentive for entity participation. The immutability and decentralised features of the Blockchain system leverage the federated-like averaging of synthetically generated data samples that generate incentives for the participation of entities in such a training setup. BlockFITS can be practically deployed in future ITS systems to improve the autonomous driving system, pedestrian safety, and vehicular object detection or more due to its model-constraint-free characteristics which provide access to a synthetic and global data whilst maintaining data privacy.

B. Kansra · A. Khanna · D. Gupta
Maharaja Agrasen Institute of Technology, New Delhi, India
e-mail: ashishkhanna@mait.ac.in

D. Gupta
e-mail: deepakgupta@mait.ac.in

H. Diddee
Bharati Vidyapeeth's College of Engineering, New Delhi, India
e-mail: harshita.bvcoend@bvp.edu.in

T. H. Sheikh
Government Degree College, Poonch, India

J. J. P. C. Rodrigues
Federal University of Piauí (UFPI), Teresina, PI, Brazil
e-mail: joeljr@ieee.org

Instituto de Telecomunicações, Aveiro, Portugal

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

253

A. Khanna et al. (eds.), *International Conference on Innovative Computing and Communications, Advances in Intelligent Systems and Computing 1388*,
https://doi.org/10.1007/978-981-16-2597-8_21

Keywords Blockchain • Intelligent transport systems • Internet of vehicular things • Federated learning • Data augmentation

1 Introduction

Given the paucity of covariate data in IoVT systems, developing a trustworthy and decentralised technique to augment local data can be massively useful in enhancing the robustness of these systems. The deep learning models used in such systems are data hungry and benefit from the provision of data obtained from different sources. It is also generally observed that the performance of deep learning models increases exponentially, if more representative data is provided at the training stage. The data used in IoVT systems is generally acquired using mounted cameras, sensors, embedded global positioning systems (GPS), Light Detection & Ranging (LIDAR) sensors, accelerometers, optical sensors, etc. As with other domains of deep learning, augmentation of data derived through these sources can improve the performance of such systems significantly, especially since, a large of amount of data being used in these systems is real time and local context dependent. These sources can collect important useful data in real time which can be used to train locally deployed models with applications in autonomous driving, vehicles detection, pedestrian detection, traffic signs recognition, lane detection, etc. [20]. Given the dependence of IoVT systems on such dynamically changing data, it seems natural to seek a method which can effectively use locally collected data points via onboard sensors, to train robust deep learning models for IoVT use cases. Generally in the given, the data collected via local sensors is pre-processed, augmented, and then an edge deployed model is trained on the ITS entity itself. This limits the utilisation of data collated by the entity since the neighbouring entities obviously do not have access to the data collected by entities individually. If this data were to be shared with a central authority, the possibility of data breaches and malicious manipulation endanger the safe and efficient storage of such data [14]. Hence, the need to keep data private as well as shareable while not depending on a single central organisation, calls for the use of a distributed-ledger enabled system which can alleviate the problem of having a single point of failure/control, as well as provide incentives to the participatory entities using an ledger-enabled application such as Blockchain. Additionally, the need to train a set of locally deployed models in a privacy preserving manner encourages the use of a decentralised and anonymized method of collaborative model training—Federated Learning. Lastly, Under collaborative and distributed learning systems—the lack of a proper incentive provision may discourage entities from participating in cooperative learning. To counter this, several incentive strategies have been explored, even in the context of ITS systems; the additional advantage of using distributed ledgers ensure that workers are voluntarily joining the network and no single company overhead costs are involved in it hence making it a decentralised system (Figs. 1 and 2). In accordance with the following principle motivations, we define BlockFITS with the aim to provide the following contributions:

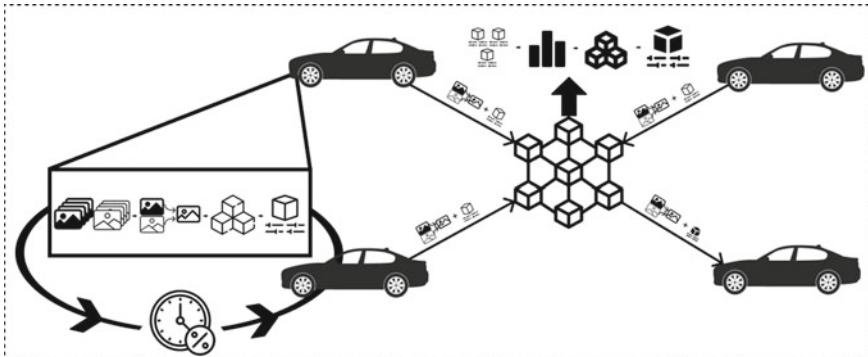


Fig. 1 Distributed ITS entities connected to BlockFITS IoVT system

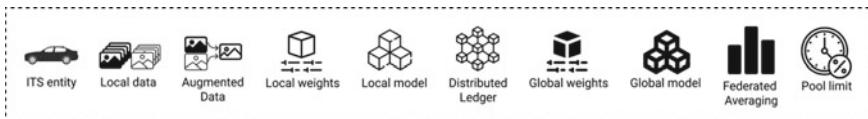


Fig. 2 Notation of the illustrated BlockFITS Model

- To tackle the paucity of quality and quality data in IoVT systems, we propose a privacy preserving data sharing technique that uses synthetically augmented to enhance the quality of the locally acquired data with ITS entities.
- A federated learning enabled distributed-ledger mechanism which allows a decentralised network of all ITS entities to share weights of their locally trained model to allow entities on the ledger to leverage the learning of the local models hosted by other entities.
- To encourage participation in such a collaborative method of learning, we specify an incentive mechanism that defines the utility function of each entity with its local accuracy; based on its performance, the entity is given a proportionate “quota” of augmentation to compute—thus allowing relatively better performing entities to reduce their computational strain due to the added data augmentation step.

2 Literature Review

2.1 Internet of Vehicular Things (IoVT)

Since the last few decades Intelligent Transport Systems(ITS) have been an important nexus for the next generation automated world. The availability of cost effective as well as reliable sensors networks gave a boost to Internet of Things (IoT) [16]. Internet of Vehicular Things (IoVT) is an application of the Internet of Things (IoT)

Table 1 A comparison between BlockFITS and other blockchain-based IoVT systems

Ref.	Objective	Privacy	Incentive	Data
[9]	Vehicular communication management	Conditional	Mission-Based Reward	Broadcasted between vehicles
[10]	Privacy preserving carpooling	Conditional	None	Encrypted and sent to a central server
[26]	Vehicular data sharing	Complete	Data Coins Rewards	Shared and stored on a private BlockChain
This	Vehicular Model and Augmented data sharing	Complete	Accuracy	Store locally and Share only Augmented Data

which focuses on making human intervention to a minimal and seamless extent whilst making vehicles more intelligent and connected. It does so by exchanging information, increasing reliability and efficiency and maintaining safety of the end user [1].

While exploring the current research about security in IoT systems it is observed that the major focus is in evolving techniques to maintain a physical level safety of the end user but we want to focus on less investigated area of safety in domain of maintaining end users privacy and keep the user data secure and still using it to create a more intelligent system [5, 6, 13] (Table 1).

2.2 *Blockchain*

Blockchain is a collection of congruent blocks with each block containing a cryptographic hash value of the last block. By design, a Blockchain is impervious to change data once written on the block. Blockchain is widely used as a distributed-ledger system managed via a P2P (Peer2Peer) network of nodes while adhering to a predefined protocol for connectivity between blocks [17]. In recent work in the field of Blockchain-based Vehicular systems there have been several consensus algorithms to achieve different objectives. In [9, 25] the authors propose a vehicle communication management system which uses practical Byzantine Fault Tolerance(pBFT) consensus. The authors of [10] propose a privacy preserving carpooling system which is a permissioned and decentralised network based on PoS consensus. The authors in [26] propose a Vehicular ad-hoc network (VANET) which leverages Consortium and decentralised network with main objective to share data and storage between vehicle using a pBFT consensus.

2.3 *Federated Learning*

Modern ITS entities have potential to access abundance of data for deep learning models which can drastically improve the user interaction and experience. However, this widely accessible data is often user privacy sensitive, and available in billions of independent entities. Federated Learning advocates on keeping the private user data on distributed entities itself, otherwise logged into a centralised data centre, and learns a shared model by aggregating the locally determined updates[12]. Most of the recent publications [3, 7, 11, 23] about application of federated learning in vehicular systems are based on aggregation of collected local weights on a centralised server; this is a partially centralised approach and solely focuses on securing the user data but lacks in providing an incentive to the participating entities which makes it hard to predict if the contributions of the participating nodes would be sustained.

2.4 *Data Augmentation*

The synthetic generation of data for several deep learning tasks has resulted in enhanced performance; [19] provides an overview which aims to validate the same in the context of time series use cases. The data collected from mounted camera on an ITS entity provides only a small set of local information ergo leads to lack of accurate judgment at global scenario. This calls for a way of introducing more data for training the system. Data Augmentation is one of the technique which can be used to do the same without the need of burdening the entity to generate more data. Data augmentation of spatiotemporal data collected in ITS entities can be augmented using by various methods like image stitching techniques proposed by [18] or using comotion algorithm to land-points from adjacent cameras, and construct a homography matrix constructed as done in [21].

3 System Model

In this section, we first delineate the definitions and notations that will be used to describe the working of the proposed, V2B2V IoVT model. Following which we explain the detailed working of the system along with the necessary implementation details. The system working is broadly divided into 2 broad sections—The first one is specific to the on-device computation and model training on the ITS entity and the subsequent transfer of the augmented data and the local weights to a ledger. The second section follows to define the pooling and aggregation of the locally derived data on a randomly selected ledger node and then its subsequent distribution on the participating entities using the specified incentive mechanism.

3.1 Requisite Definitions

In this section we attempt to explicate the frequently used terms and notation used in our system model description:

- **Local Data or LD:** This defines the set of real-time data samples collected by the ITS entity. This data may be acquired through any of the sources specified in Section I.
- **Augmented data or LD^{*}:** This defines the set of synthetically augmented set of data samples that will be generated by the ITS Entity.
- **Local Model:** Leveraging the collaborative model training traits of federated learning, the ITS entity may host an edge deployed local model; this model may be a secondary utility model for the ITS entity, such as traffic flow prediction, travel time estimation, multiple trajectory prediction, and congestion control prediction model for a wireless daisy chain network.
- **Local weights:** Local weights are the learnable parameters of the on-device deep learning model; this model is trained on LD and LD^{*}.
- **Pool limit or α :** Pool limit is the minimum number of augmented data samples that the entity must generate before it can relay this set to the ledger. The motivation of introducing this term derives from the fact that due to computational constraints on the ITS entity and the practical constraints on the constant relaying of information across to a ledger due to network constraints—the synthetic data at each ITS entity will not be relayed to a common ledger until it acquires a certain minimum number of data samples, i.e. the pool limit number of samples.
- **Pooling Frequency or λ :** This refers to the frequency at which augmented data, LD^{*}, is pooled at a commonly decided ledger node.
- **Image Stitching:** Image stitching refers to the concatenation of images, similar to the generation of panoramic image generation, using local data samples. This type of augmentation is carried out to specifically eliminate the identification and consequent reporting of duplicate objects in the same image [18]
- **Idle Time:** Since the assignment of computational resources during the operation of the ITS Entity for the task of data augmentation may not be the optimal choice bearing in mind the low priority of the data augmentation in comparison to the primary tasks of the ITS entity (including resource assignment to any deep learning model hosted locally that may be crucial to navigation, destination or traffic prediction model)—The augmentation for the proposed model will be carried out when the entity is not involved in any other resource-intensive activity—mostly when the entity is non-operational for a significantly long period of time such as when it is parked or being charged [2].

3.2 Workflow of the System

Generation of LD* from LD: After the acquisition of local data samples—LD* is populated as follows: Augmentation manipulations such as brightness enhancements, horizontal and vertical shifts, shadow casting, flipping, and sub-sampling are already used to enhance the quantity of locally available dataset. In addition to these techniques, we use Image Stitching to generate more synthetic images. Note that this augmentation will take place during the idle time only [15].

Training of the local model: Data samples from both, LD and LD* are used to train the local model on device. Note that the decision of what proportion of augmented samples to use in the training of the local model is a hyper parameter, i.e. the training data split [between locally acquired data samples and synthetically generated/received] data samples may be modified experimentally to achieve the split that achieves the maximum local accuracy.

Transfer of weights (and LD*, if applicable) to the ledger Due to their inherent characteristic of hosting immutable data which whilst being immune to malicious manipulation, still supports the transparent viewing of the artificially augmented data that is generated by each node—ledgers provide an ideal platform for the established setup. The weights of the local model are relayed to the ledger node after running a predefined set of epochs on the local models (The number of epochs being run on the local model is also a hyperparameter). This step will be accompanied by the relaying of LD* as soon as the number of samples in LD > λ . Note that the relayed LD* may be directly transferred to the pooling node [4] (Fig. 3).

Aggregation and Redistribution of local weights Since there is no central supervising entity in such a system, the aggregation of weights hosted at all ledger nodes is carried out on a randomly selected node in the ledger that is referred to as the pooling node. This aggregation, done in accordance with the FedAVG algorithm [12], is done without any additional pre-processing, this aggregation may be carried out using a smart contract as well. To test the performance of these aggregated and averaged weights, the aggregating node relays these aggregated nodes to the ITS entity it is connected to, which computes the accuracy of the model on its local test set. If the loss assumes a converging trend, the ledger distributes these aggregated weights to all the other entities on the network.

Establishing Incentive-Based Distribution of pooled Data to the entities Similar to the fashion of redistribution of local weights, the LD* hosted at each node are pooled at a randomly selected node. This pool of synthetically generated samples is shuffled and relayed to all the nodes on the ledger, which are finally relayed to the ITS entities. This redistribution of samples is governed by an incentive scheme which operates as follows: The scheme is inspired majorly by the Individual Profit Sharing Scheme proposed by [22], which maps the utility function of the scheme as the local accuracy of each ITS Entity (Eq. 1):

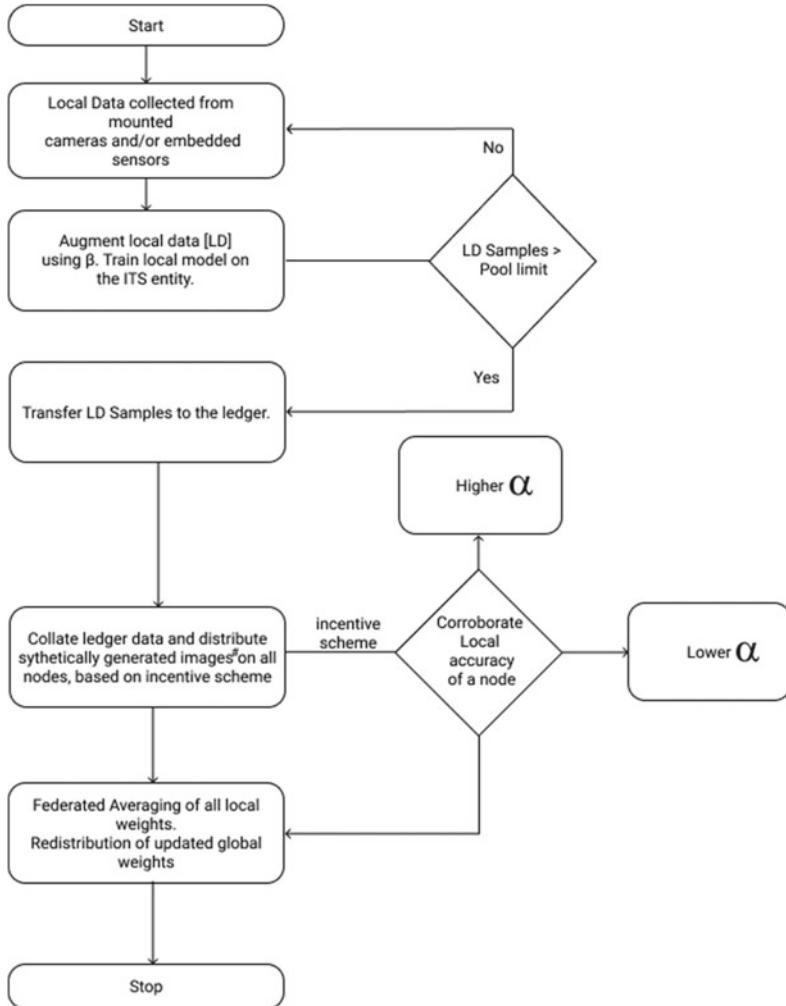


Fig. 3 Flowchart depicting the cycle of BlockFITS

$$\begin{aligned}
 U_i(t) &= V_i \times \alpha \\
 0 \leq V_i &\leq 1 \\
 \alpha &\in \mathbb{R}
 \end{aligned} \tag{1}$$

In essence, Higher the local accuracy of the system, higher is the utility of the contributing entity. Using a smart contract, the local accuracy of all the ITS entities can be related to their utilities. The utility of the entity decides how much of its resources would it be asked to sacrifice during the generation of LD* in the next iteration, i.e. a high utility entity will receive an updated α or pooling limit—which

will allow it to relay its LD* without having augmented a LD* commensurate to its poorer performing peers on the network. This ensures that high performing entities enjoy the access to the collaborative training as well as the diverse pool of synthetic data without compromising on their computational resources consistently, which will be necessary if they are too augment LD* from LD [8, 24].

4 Conclusion

In this paper, we present BlockFITS, A federated learning-based, Blockchain-enabled data augmentation system that allows participating ITS entities to leverage a rich set of locally gathered yet artificially pruned data to collaboratively train their deep learning models. Unlike most existing systems—BlockFITS establishes an incentive mechanism focused around the primary factor that affects local model accuracy, i.e. the data that an entity uses. Moreover, it attempts to give consideration to the practical idea that all entities participating in the collaborative training do not have massive resource capabilities and hence, must be advantaged, if their contributions benefit the network at large. This work must be further analysed to identify and mitigate the caveats that arise from hosting the synthetic data on a Blockchain node. Additionally, the incentive mechanism may be enhanced to include other data quality-driven metrics such as one that accounts for the ITS entity that provides the set of data with the highest sample diversity or the entity that provides data that is representative of some temporal trend.

Acknowledgements We are very grateful for the partial funding provided by FCT/MCTES through national funds and when applicable co-funded EU funds under the Project UIDB/50008/2020, and by Brazilian National Council for Scientific and Technological Development—CNPq, via Grant No. 309335/2017-5. This work would not have been possible without this support.

References

1. S. Chavhan, D. Gupta, S. Garg, A. Khanna, B.J. Choi, M.S. Hossain, Privacy and security management in intelligent transportation system. *IEEE Access* **8**, 148677–148688 (2020)
2. W. Chmiel, J. Dańda, A. Dziech, S. Ernst, P. Kadłuczka, Z. Mikrut, P. Pawlik, P. Szwed, I. Wojnicki, Insigma: an intelligent transportation system for urban mobility enhancement. *Multimedia Tools and Applications* **75**(17), 10529–10560 (2016). (Sep)
3. Z. Du, C. Wu, T. Yoshinaga, K.A. Yau, Y. Ji, J. Li, Federated learning for vehicular internet of things: Recent advances and open issues. *IEEE Open Journal of the Computer Society* **1**, 45–61 (2020)
4. A. Goel, A. Agarwal, M. Vatsa, R. Singh, N. Ratha, Deepring: protecting deep neural network with blockchain, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2019), pp. 2821–2828
5. L. Janušová, S. Čermancová, Improving safety of transportation by using intelligent transport systems. *Proc. Eng.* **134**(12), 14–22 (2016). <https://doi.org/10.1016/j.proeng.2016.01.031>

6. L. Janušová, S. Čermancová, Improving safety of transportation by using intelligent transport systems. *Proc. Eng.* **134**, 14–22 (2016). <https://doi.org/10.1016/j.proeng.2016.01.031>, <http://www.sciencedirect.com/science/article/pii/S1877705816000345>, In *Transbaltica 2015: Proceedings of the 9th International Scientific Conference* 7–8 May 2015. Vilnius Gediminas Technical University, Vilnius, Lithuania
7. J. Kang, Z. Xiong, D. Niyato, S. Xie, J. Zhang, Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal* **6**(6), 10700–10714 (2019)
8. J. Kang, Z. Xiong, D. Niyato, D. Ye, D.I. Kim, J. Zhao, Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory. *IEEE Transactions on Vehicular Technology* **68**(3), 2906–2920 (2019)
9. L. Li, J. Liu, L. Cheng, S. Qiu, W. Wang, X. Zhang, Z. Zhang, Creditcoin: A privacy-preserving blockchain-based incentive announcement network for communications of smart vehicles. *IEEE Transactions on Intelligent Transportation Systems* **19**(7), 2204–2220 (2018)
10. M. Li, L. Zhu, X. Lin, Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing. *IEEE Internet of Things Journal* **6**(3), 4573–4584 (2019)
11. Y. Lu, X. Huang, Y. Dai, S. Maharjan, Y. Zhang, Federated learning for data privacy preservation in vehicular cyber-physical systems. *IEEE Network* **34**(3), 50–56 (2020)
12. H.B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. Arcas, Communication-efficient learning of deep networks from decentralized data (2016)
13. M.A. Regan, J.A. Oxley, S.T. Godley, C. Tingvall, Intelligent transport systems: safety and human factors issues No. 01/01 (2001)
14. F. Sakiz, S. Sen, A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov. *AdHoc Netw.* **61**(03) (2017). <https://doi.org/10.1016/j.adhoc.2017.03.006>
15. C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning. *J. Big Data* **6**(1), 60 (2019) (Jul). <https://doi.org/10.1186/s40537-019-0197-0>
16. A. Śladkowski, W. Pamuła, *Intelligent Transportation Systems-Problems and Perspectives*, vol. 303 (Springer, 2016)
17. M. Swan, *Blockchain: Blueprint for a New Economy* (O'Reilly Media, Inc. 2015)
18. P. Tsao, T.U. Ik, G.W. Chen, W.C. Peng, Stitching aerial images for vehicle positioning and tracking (11), 616–623 (2018). <https://doi.org/10.1109/ICDMW.2018.00096>
19. Q. Wen, L. Sun, X. Song, J. Gao, X. Wang, H. Xu, Time series data augmentation for deep learning: A survey (2020)
20. W. Wu, Z. Yang, K. Li, Internet of Vehicles and applications (12), 299–317 (2016). <https://doi.org/10.1016/B978-0-12-805395-9.00016-2>
21. Y. Wu, C. Liu, S. Lan, M. Yang, 3d road scene monitoring based on real-time panorama. *J. Appl. Math.* **2014**, 403126 (2014). <https://doi.org/10.1155/2014/403126>
22. G. Yang, S. He, Z. Shi, J. Chen, Promoting cooperation by the social incentive mechanism in mobile crowdsensing. *IEEE Communications Magazine* **55**(3), 86–92 (2017)
23. D. Ye, R. Yu, M. Pan, Z. Han, Federated learning in vehicular edge computing: A selective model aggregation approach. *IEEE Access* **8**, 23920–23935 (2020)
24. K. Yeow, A. Gani, R.W. Ahmad, J.J.P.C. Rodrigues, K. Ko, Decentralized consensus for edge-centric internet of things: A review, taxonomy, and research issues. *IEEE Access* **6**, 1513–1524 (2018)
25. Y. Yuan, F. Wang, Towards blockchain-based intelligent transportation systems, in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)* (2016), pp. 2663–2668
26. X. Zhang, X. Chen, Data security sharing and storage based on a consortium blockchain in a vehicular adhoc network. *IEEE Access* **PP**(01), 1–1 (2019). <https://doi.org/10.1109/ACCESS.2018.2890736>

Comparative Analysis for Improving Accuracy of Image Classification Using Deep Learning Architectures



Gopal Sakarkar^{ID}, Ketan Paithankar, Prateek Dutta, Gaurav Patil, Shivam, Ruchi Chaturvedi, Akshita Bhimarapu, and Riddhi Mandal

Abstract Image classification is a classic problem in areas pertaining to Computer Vision, Image Processing, and Machine Learning. This paper aims to compare the various Deep Learning Architectures to improve the accuracy of Image Classification to select the best Deep Learning Architecture by implementing and testing various Deep Learning Architectures in combination with Dense Neural Networks. This comparative study helps to improve the accuracy of image separation in both training and testing databases. For targeted training and testing, 3000 training images and 1000 test images were used. The result of the Deep Learning-based classification of images using the platform as Google Colab showed how accurate classification was done by comparing various deep learning architectures.

Keywords Deep learning · Binary image classification · TensorFlow · Transfer learning · Machine learning · Architectures

1 Introduction

The task of extracting informative classes from a multiband raster image is called image classification. Supervised image classification techniques were used in our paper. Many initiatives were taken by the researcher in the field of image classification [1–3].

G. Sakarkar (✉) · P. Dutta · G. Patil · Shivam · R. Chaturvedi · A. Bhimarapu · R. Mandal
Department of Artificial Intelligence, G H Raisoni College of Engineering, Nagpur, India
e-mail: gopal.sakarkar@raisoni.net

K. Paithankar
Chief Technology Officer, konverge.ai, Nagpur, Maharashtra, India
e-mail: ketan@konverge.ai

1.1 Motivation

The challenge of image separation of Cats and Dogs relied on the CAPTCHA challenge from the Dogs versus Cats competition, Kaggle [4]. It is easier for a human to classify Cats and Dogs but Cats and Dogs are more complex to distinguish by default.

Constructing Deep Learning classifiers to address this problem is one of the topics where many people are working. Based on a combination of Color and Texture features, using the SVM classifier Golle finds out the accuracy of 82.7%. The best results obtained by Golle were obtained using 10000 images from the data, 8000 for training and 2000 for testing [5].

1.2 Problem Identification

Our basic task is the comparison of architectures to classify the image into a Cat or a Dog. Images of dogs and cats from the training dataset are the input for the model, while the output is the classification of the image into a Cat versus Dog.

The Cats and Dogs dataset is from the competition provided by well-known dataset provider Kaggle. Our training dataset contains 3000 images, which includes 1500 images of dogs and 1500 images of cats, while the test dataset contains 1000 images. The dimension of these images is 150×150 . The basic aim of this research work is to train a classification model to classify cats and dogs from images.

As shown in Fig. 1, it can be observed that the images from the training dataset are the input for the learning task and the trained classification model is the output.

To classify images from the test dataset, the classification model trained over the training dataset is used. The evaluation of the classification accuracy over the different deep learning architectures is calculated over the test data. Images from the test dataset are the input, and the labeling is the output, as shown in Fig. 2.

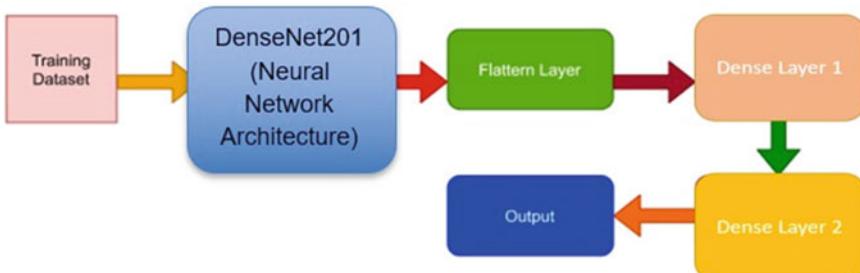


Fig. 1 Architecture for learning task

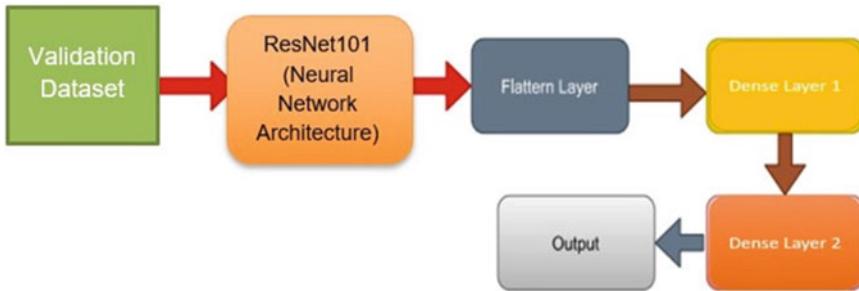


Fig. 2 Architecture for performance task

DenseNets has several compelling benefits: it reduces the problem of gradient disappearance, enhances feature distribution, promotes feature utilization, and significantly reduces the number of parameters.

Flatten is a layer that converts a map of an integrated element into a single column transferred to a fully connected layer. A dense layer can add a layer that is fully connected to the neural network. The thick final layer and function of sigmoid activation separate the image as a dog and a cat.

1.3 Solution

Based on the features of the image, images are recognized using different Neural Networks architecture in deep learning. To solve the complexity faced during the conventional methods, total feature extraction model is created. From the training set of images, the extractor of the integrated model should be able to learn how to extract and differentiate the features accurately [6]. In this solution, different architectures of neural networks are applied to increase the accuracy of image classification. We have used Architectures like InceptionV3, ResNet101, ResNet101V2, ResNet152, ResNet152V2, ResNet50, ResNet50V2, VGG16, VGG19, DenseNet121, DenseNet169, DenseNet201, EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4, EfficientNetB5, EfficientNetB6, EfficientNetB7, InceptionResNetV3, MobileNet, MobileNetV2, and Xception [7] along with dense neural networks for this purpose.

Making use of the supervised image classification technique, we are classifying the images of Cats vesus Dogs, which has been taken from Kaggle's competition. By the comparative study of 24 different deep learning architectures, we have trained the classification model to classify Cats and Dogs from images.

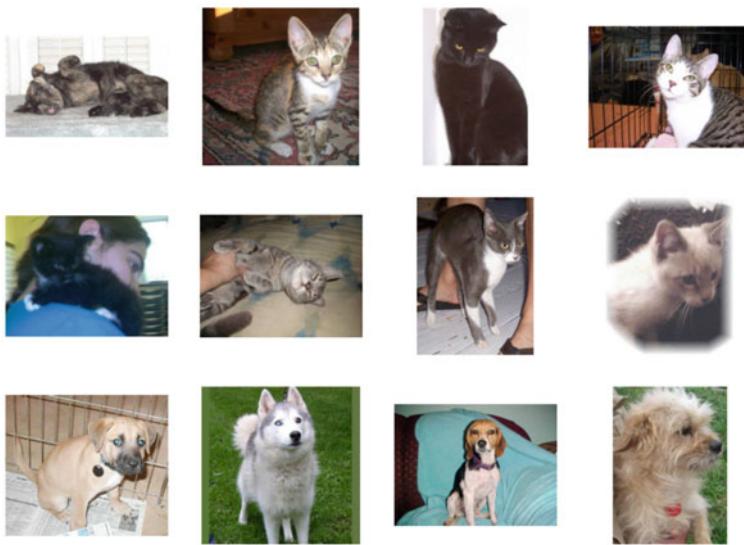


Fig. 3 Sample images from the dataset

2 Dataset

The dataset in Kaggle Cats versus Dogs classification is divided into folders for each class. The dataset is divided into training and testing sets. Two folders one for cat images and the other for dog images are composed of the training and testing sets, with random 1000 images of each cat and dog for training and 500 images of each cat and dog for validation. The images are of various shapes and sizes, but to train a CNN, the images should be of the same size. Figure 3 shows some sample images used, from the dataset.

3 Related Architecture Work

DenseNet concatenates the output of previous layers to the long run layer. DenseNet has many versions like DenseNet121, DenseNet169, DenseNet201. The varieties within the versions contain the various number of layers used. The layers associated with DenseNet area were Convolutional and Pooling layers, Transition Layers, Classification layers and Dense Block [8].

InceptionResNet came into existence after the success of ResNet design. It galvanized to form a hybrid origination model. It additionally has completely different versions named InceptionResNetv1, InceptionResNetv2, and InceptionResNetv3. Our study has proceeded with InceptionResNetv3 [9].

ResNet: to enhance accuracy and performance, some further layers are unit-stacked in Deep Neural Networks. The motivation for adding these layers is that adding more layers finds out more accurate output and reduces loss. Taking associate degree examples of image recognition, the first layer identifies edges, the second identifies textures, the third layer identifies objects, and so on. However, it's been found that there's the best threshold for depth with the normal convolutional neural network model. The feature of ResNet referred to as skip connections solves the matter of vanishing gradient in DNN by giving a brief path for the gradient to flow through [10].

EfficientNets as a first it divides the initial convolution into two stages of Depthwise Convolution and Pointwise convolution to reduce the cost of calculation and provide minimum loss of accuracy. Then, there is a layer that first extends the channels and then squeezes them to skip layers with few channels, and then it uses linear activation in the last layer of each block to prevent loss occurred due to ReLU. The EfficientNet model group has 8 models starting from B0 to B7 where the number refers to different versions with more parameters and higher accuracy. We have used the entire B0 to B7 model groups for our study [11].

VGG Net: It exploits the Convolutional Neural Network where the embedded images cleverly measure a series of layers, i.e., convolutional, pooling, flattening, and fully connected layers, and then remove the CNN image separating image. If the CNN models were built from scratch, then it needs to fine-tune the model by exploiting the process of adding images. As a result, our paper makes use of one of the selected Models—VGG-16—to separate the image and check the accuracy of the training information and verification information [12].

InceptionV3 is an architecture of CNN which is built of 48 layers [13]. It has a pre-trained version that we used because it is already trained with millions of images from the ImageNet database, thus is able to classify animals efficiently (Table 1).

MobileNet is a type of factorized convolution that is based on depthwise separable convolution. The specialty of MobileNet is its requirement of less computational power to run or apply learning to it. This is the reason it is perfect for machines without GPU or low computational efficiency. The MobileNet architecture has 3 versions named MobileNetV1, MobileNetV2, and MobileNetV3. We have used MobileNetV1 and MobileNetV2 versions which use 53 different layers for classification. MobileNetV2 are much faster than MobileNetV1 for the same [14].

All this Neural Net architecture mentioned in the Table 1, one by one was used as an input to the Flatten layer followed by Dense layers and an output dense layer.

4 Result Analysis

Table 2 shows the comparison of training and validation loss with the maximum and minimum values among its architectures, also, the training and validation accuracy versus the minimum and maximum values among the architecture.

Table 1 Architecture and their training and testing versus accuracy and loss

S. No.	Architectures	Loss	Val_loss	Accuracy	Val_accuracy
1	Xception [7]	0.0162	0.4835	0.996	0.955
2	DenseNet121 [8]	0.0195	0.3614	0.9955	0.952
3	DenseNet169[8]	0.0098	0.2186	0.9670	0.975
4	DenseNet201 [8]	0.0049	0.2098	0.9990	0.977
5	InceptionResNetV3 [9]	0.02	327.502	0.996	0.939
6	ResNet101 [10]	0.0181	5.3517	0.9170	0.9980
7	ResNet101V2 [10]	0.0115	0.697	0.9965	0.942
8	ResNet152 [10]	0.3613	0.7564	0.994	0.923
9	ResNet152V2 [10]	0.0943	3.0634	0.9905	0.87
10	ResNet50 [10]	0.0124	30.3982	0.997	0.795
11	ResNet50V2 [10]	0.0116	3.3807	0.9975	0.877
12	EfficientNetB0 [11]	0.0287	1.6725	0.9935	0.643
13	EfficientNetB1 [11]	0.0151	0.9299	0.9965	0.765
14	EfficientNetB2 [11]	0.0248	2.1877	0.9935	0.71
15	EfficientNetB3 [11]	0.0472	1.5623	0.997	0.795
16	EfficientNetB4 [11]	0.0175	0.5629	0.997	0.838
17	EfficientNetB5 [11]	0.0316	0.67	0.9935	0.923
18	EfficientNetB6 [11]	0.0185	0.3903	0.9955	0.926
19	EfficientNetB7 [11]	0.0322	0.8246	0.9945	0.907
20	VGG16 [12]	0.0825	0.3183	0.981	0.924
21	VGG19 [12]	0.0616	0.7166	0.9795	0.901
22	InceptionV3 [13]	0.0102	0.3815	0.9975	0.945
23	MobileNet [14]	0.0201	0.2758	0.9955	0.967
24	MobileNetV2 [14]	0.017	3.3273	0.997	0.878

Table 2 Max Min versus training and validating-loss and accuracy

	Max	Min
Training loss	Resnet152 (0.3613)	DenseNet201 (0.0049)
Validation loss	InceptionresNetV3 (327.5601)	DenseNet201 (0.2098)
Training accuracy	DenseNet201 (0.9990)	DenseNet169 (0.9670)
Validation accuracy	ResNet101 (0.9980)	EfficientNetB0 (0.643)

5 Model Compilation, Training, and Model Evaluation

While running model compilation, different parameters like loss function, optimizer algorithm, and the list of metrics need to be focused on. RMSprop is used as an optimization algorithm. Binary Cross Entropy is used as a function of the loss and accuracy by the end metric used. Sequential model and the fit () are used for model training. The model is trained for 15 epochs. Below is Fig. 4 showing the end result of training, i.e., trains accuracy, validation accuracy, train loss, validation loss, and epochs for the best architecture in terms of highest accuracy on the training dataset.

The Graph shown in Fig. 5 represents the relation between increasing epoch

```
Found 2000 images belonging to 2 classes.
Found 1000 images belonging to 2 classes.
Epoch 1/15
100/100 - 33s - loss: 0.2867 - accuracy: 0.9185 - val_loss: 0.6385 - val_accuracy: 0.9130
Epoch 2/15
100/100 - 30s - loss: 0.0896 - accuracy: 0.9720 - val_loss: 0.1194 - val_accuracy: 0.9810
Epoch 3/15
100/100 - 30s - loss: 0.0528 - accuracy: 0.9850 - val_loss: 0.1773 - val_accuracy: 0.9750
Epoch 4/15
100/100 - 30s - loss: 0.0414 - accuracy: 0.9865 - val_loss: 0.1527 - val_accuracy: 0.9810
Epoch 5/15
100/100 - 30s - loss: 0.0459 - accuracy: 0.9885 - val_loss: 0.2291 - val_accuracy: 0.9680
Epoch 6/15
100/100 - 30s - loss: 0.0476 - accuracy: 0.9875 - val_loss: 0.1451 - val_accuracy: 0.9700
Epoch 7/15
100/100 - 30s - loss: 0.0186 - accuracy: 0.9955 - val_loss: 0.2901 - val_accuracy: 0.9660
Epoch 8/15
100/100 - 30s - loss: 0.0221 - accuracy: 0.9945 - val_loss: 0.2176 - val_accuracy: 0.9690
Epoch 9/15
100/100 - 30s - loss: 0.0222 - accuracy: 0.9950 - val_loss: 0.2071 - val_accuracy: 0.9700
Epoch 10/15
100/100 - 30s - loss: 0.0183 - accuracy: 0.9945 - val_loss: 0.2285 - val_accuracy: 0.9730
Epoch 11/15
100/100 - 30s - loss: 0.0235 - accuracy: 0.9945 - val_loss: 0.2369 - val_accuracy: 0.9660
Epoch 12/15
100/100 - 30s - loss: 0.0228 - accuracy: 0.9955 - val_loss: 0.4937 - val_accuracy: 0.9580
Epoch 13/15
100/100 - 30s - loss: 0.0244 - accuracy: 0.9960 - val_loss: 0.2256 - val_accuracy: 0.9740
Epoch 14/15
100/100 - 30s - loss: 0.0130 - accuracy: 0.9975 - val_loss: 0.2515 - val_accuracy: 0.9770
Epoch 15/15
100/100 - 30s - loss: 0.0049 - accuracy: 0.9990 - val_loss: 0.2098 - val_accuracy: 0.9770
```

Fig. 4 Loss, accuracy, validation accuracy and validation loss of denseNet 201 till Epoch 15

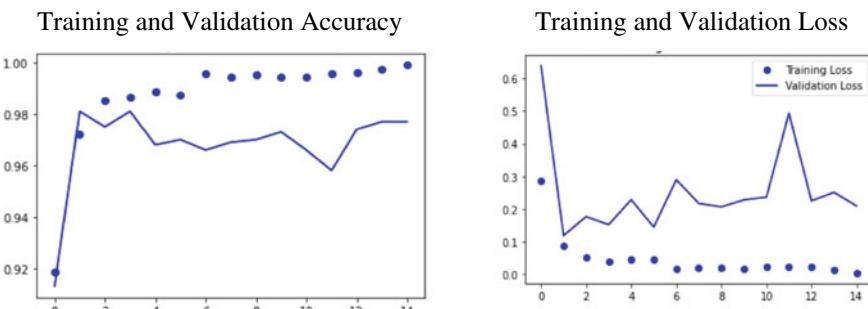


Fig. 5 Relation between increasing epoch and loss/accuracy on train/test data

and loss/accuracy on train/validation data for DensNet 201. The X-axis represents the epoch. There are 15 epochs numbered from 0 to 14. The Y-axis represents the accuracy/loss ranging from 0 to 1. The relation between accuracy/loss and epoch for validation data is represented by a blue continuous line. The relation between accuracy/loss and epoch for training data is represented by blue distributed dots.

It is evident from Fig. 5 that the train accuracy kept on increasing with the epoch number and the accuracy of validation also increases with the epoch number. Figure 5 shows the train loss and validation loss versus the number of epochs on DenseNet201.

It is evident from Fig. 4 that the train loss kept decreasing with the number of epochs, also the validation loss decreased with the increasing epochs. The final accuracy of 99.90% was obtained on the training data epoch 15 for DenseNet201.

Below is Fig. 6 showing the end result of training, i.e., the accuracy of training, accuracy of validation, loss of train, loss of validation, and epochs for the best architecture in terms of highest accuracy on the validation dataset.

The Graph shown in Fig. 7 represents the relation between increasing epoch and loss/accuracy on train/validation data for ResNet 101. The X-axis represents the epoch. There are 15 epochs numbered from 0 to 14. The Y-axis represents the accuracy/loss ranging from 0 to 1. The relation between accuracy/loss and epoch for validation data is represented by a blue continuous line. The relation between accuracy/loss and epoch for training data is represented by blue distributed dots.

```
Found 2000 images belonging to 2 classes.
Found 1000 images belonging to 2 classes.
Epoch 1/15
100/100 - 37s - loss: 0.5083 - accuracy: 0.9020 - val_loss: 8.7354 - val_accuracy: 0.5000
Epoch 2/15
100/100 - 35s - loss: 0.0885 - accuracy: 0.9800 - val_loss: 668.0923 - val_accuracy: 0.5000
Epoch 3/15
100/100 - 35s - loss: 0.0630 - accuracy: 0.9885 - val_loss: 4581.4609 - val_accuracy: 0.5000
Epoch 4/15
100/100 - 36s - loss: 0.0189 - accuracy: 0.9960 - val_loss: 184.5970 - val_accuracy: 0.5010
Epoch 5/15
100/100 - 35s - loss: 0.0691 - accuracy: 0.9900 - val_loss: 159.0669 - val_accuracy: 0.5010
Epoch 6/15
100/100 - 35s - loss: 0.2024 - accuracy: 0.9915 - val_loss: 12.0045 - val_accuracy: 0.5020
Epoch 7/15
100/100 - 36s - loss: 0.0162 - accuracy: 0.9950 - val_loss: 3.3838 - val_accuracy: 0.7320
Epoch 8/15
100/100 - 35s - loss: 0.0253 - accuracy: 0.9935 - val_loss: 0.6990 - val_accuracy: 0.7900
Epoch 9/15
100/100 - 35s - loss: 0.0067 - accuracy: 0.9980 - val_loss: 0.9509 - val_accuracy: 0.9280
Epoch 10/15
100/100 - 35s - loss: 0.0226 - accuracy: 0.9950 - val_loss: 1.0587 - val_accuracy: 0.9330
Epoch 11/15
100/100 - 35s - loss: 0.0311 - accuracy: 0.9945 - val_loss: 3.7365 - val_accuracy: 0.9290
Epoch 12/15
100/100 - 35s - loss: 0.0069 - accuracy: 0.9980 - val_loss: 1.6743 - val_accuracy: 0.8980
Epoch 13/15
100/100 - 36s - loss: 0.0525 - accuracy: 0.9970 - val_loss: 2.3398 - val_accuracy: 0.9290
Epoch 14/15
100/100 - 35s - loss: 0.2148 - accuracy: 0.9975 - val_loss: 0.7422 - val_accuracy: 0.9440
Epoch 15/15
100/100 - 35s - loss: 0.0181 - accuracy: 0.9980 - val_loss: 5.3517 - val_accuracy: 0.9170
```

Fig. 6 Loss, accuracy, validation accuracy, and validation loss of ResNet 101 with Epoch 15

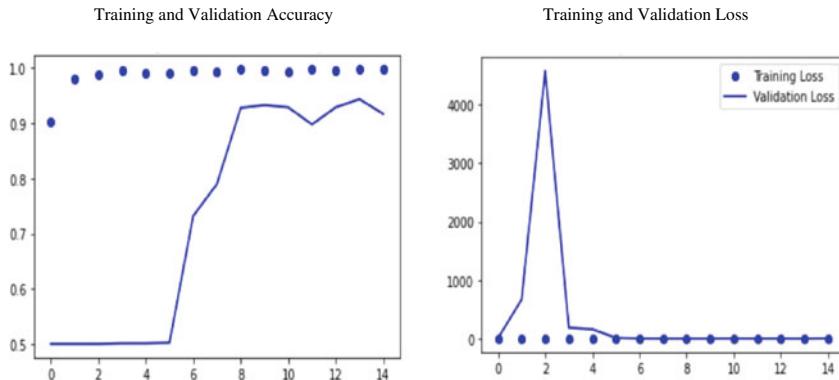


Fig. 7 Training loss kept decreasing with the number of epochs

It is evident from Fig. 7 that the training accuracy kept on increasing with epoch number and the accuracy of validation also increases with the epoch number. Figure 7 shows the training loss and validation loss versus the number of epochs on ResNet101.

It is evident from Fig. 7 that the training loss kept decreasing with the number of epochs, also the validation loss decreased with the increasing epochs. The final accuracy of 99.80% was obtained on the Validation data at epoch 15 for ResNet101.

6 Discussion and Comparison

According to the authors in [15], image classification is performed mistreatment convolutional neural network that is becoming customary ever since Alex Krizhevsky, Geoff Hinton, and Ilya Sutskevar won ImageNet in 2012. Generally, convolutional neural networks use GPU technology because of the immense variety of computations, however, they proposed a technique where we tend to build a dreadfully tiny network that can work on computer hardware additionally. The network was trained using a set of Kaggle Dog-Cat dataset. The trained classifier will classify the given image into either a cat or a dog. The identical network will be trained with the other dataset and classifies the pictures into one among the 2 predefined categories.

The comparison of accuracy for different approaches used by the different authors for image classification of Cats versus Dogs is as follows:

Method	Accuracy (%)
Golle et al. [5]	82.7
Omkar M Parkhi et al. [16]	92.9
Proposed Methodology	99.80

7 Future Scope

In this research paper, we had proposed the binary technique of classifying the images. Binary represents the number of classifying objects. Likewise, here we had used two objects only (i.e., Cat and Dog). So, in the future, multiple techniques of image classification can be done which include more than the two categorical objects to classify the feature. Other packages like Keras [17] and TensorFlow [18] and architecture can also be used in further experiments.

8 Experiment Approach

In this paper, we had used 24 different architectures to achieve a higher accuracy of image classification task and among them, we got DenseNet201 as the highest accuracy on training dataset of 99.90%, while in other papers [19–22] only a few architectures were implemented and accuracy was also low as compared to this paper.

9 Limitation and Precaution

While performing this task, you should be aware of overfitting. You need to check the accuracy of the model by setting the epoch value in such a way that it should not get overfitted.

10 Conclusion

For Data Analysis and predictions, Deep Learning is a widely used learning method. For Image Classification problems, Deep Learning is also a very popular method. The proposed architecture of Deep Learning in this paper improves the accuracy of Image Classification. We used 24 different Neural Network Architectures with a flatten layer, followed by dense layer and finally a dense layer with a sigmoid classifier. With experiments, we obtained results for each combination and observed that the training accuracy for cats and dogs was highest for DenseNet201 which is 99.90%, and validation accuracy for Cat and Dog classification was the highest for ResNet101 at the epoch 15 which is 99.80%.

Acknowledgements This research work is carried out in collaboration with IT Company Konverge.Ai Pvt. Ltd., Pune, M.H.

References

1. D. Lu, Q. Weng, A survey of image classification methods and techniques for improving classification performance. *Int. J. Remote Sens.* **28**(5), 823–870 (2007)
2. A. Vailaya, M.A.T. Figueiredo, A.K. Jain, H.J. Zhang, Image classification for content-based indexing. *IEEE Trans. Image Process.* **10**(1) (2001)
3. O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification. http://www.cs.ucf.edu/courses/cap6412/fall2009/papers/InDefenceOfNN_CVPR08.pdf
4. J. Elson, J.R. Douceur, J. Howell, J. Saul, Asirra: a CAPTCHA that exploits interest-aligned manual image categorization, in *ACM Conference on Computer and Communications Security*, vol. 7 (2007), pp. 366–374
5. P. Golle, Machine learning attacks against the Asirra CAPTCHA, in *Proceedings of the 15th ACM Conference on Computer and Communications Security* (2008), pp. 535–542
6. M. Manoj krishna, M. Neelima, M. Harshali, M. Venu Gopala Rao, *Image classification using Deep learning*
7. F. Chollet, Xception: Deep learning with depthwise separable convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 1251–1258
8. G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks; in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 4700–4708
9. A. Kapoor, R. Shah, R. Bhuvan, T. Pandit, Understanding Inception Network Architecture For Image Classification
10. Q. Jiang, The Architechture of Resnet. https://www.researchgate.net/figure/The-architecture-of-the-ResNet50-network-in-this-paper_fig2_3307448002
11. M. Tan, Q. Le, Efficientnet: rethinking model scaling for convolutional neural networks (2019). arXiv preprint [arXiv:1905.11946](https://arxiv.org/abs/1905.11946)
12. S. Tammina, Transfer learning using VGG-16 with deep convolutional neural network for classifying images. *Int. J. Sci. Res. Publ.* **9**(10), 143–150 (2019)
13. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision (2015). [arXiv:1512.00567v3](https://arxiv.org/abs/1512.00567v3) [cs.CV]. Accessed 1 Dec 2015
14. A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: efficient convolutional neural networks for mobile vision applications. [arXiv:1704.04861v1](https://arxiv.org/abs/1704.04861v1) [cs.CV] Accessed 17 Apr 2017
15. M. Shanmukhi, K.L. Durga, M. Mounika, K. Keerthana, Convolutional neural network for supervised image classification. *Int. J. Pure Appl. Math.* **119**(14), 77–83 (2018)
16. O.M. Parkhi, A. Vedaldi, A. Zisserman, C.V. Jawahar, Cats and Dogs, in *2012 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2012), pp. 3498–3505
17. <https://keras.io/>
18. <https://www.tensorflow.org/>
19. B. Liu, Y. Liu, K. Zhou, Image classification for dogs and cats. TechReport (University of Alberta, 2014)
20. M.A. Abu, N.H. Indra, A.H. Abd Rahman, N.A. Sapiee, I. Ahmad, A study on image classification, based on deep learning and tensorflow. *Int. J. Eng. Res. Technol.* **12**(4), 563–569 (2019)
21. J. Liu, F.P. An, Image classification algorithm based on deep learning-kernel function. *Sci. Program.* **2020** (2020)
22. M. Xin, Y. Wang, Research on image classification model based on deep convolution neural network. *EURASIP J. Image Video Process.* (1), 40 (2019)

Other Advance Reference

23. R. Poojary, A. Pai, Comparative study of model optimization techniques in fine-tuned CNN models, in *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)* (IEEE, 2019), pp. 1–4
24. S. Marbhal, M. Kumar, *Evaluation of Datasets for CNN based Image Classification* (2020)

Infrared Thermography-Based Facial Classification Using Machine Learning



Kumud Rani, Mala Kalra, and Rakesh Kumar

Abstract Intelligent human-computer interaction (HCI) utilizes facial information, artificial intelligence, adaptive computation, and emotional awareness to improve human interaction with machines. Hence, analysis of face expression has been one of the important mediums for understanding actions and modeling emotions. This paper has introduced an Infrared Thermography (IT) based approach on MATLAB which uses 2D Discrete Wavelet Transform (2D-DWT) for decomposition of the captured thermal images of faces of different persons followed by feature extraction. Principle Component Analysis (PCA) has been used in respect to decrease the dimension of the extracted feature vector and thereafter the selected features have been rated in order to achieve the most relevant feature vector. Lastly, all the selected features in a single feature vector have been put into the Support Vector Machine (SVM) and Artificial Neural Network (ANN) for person identification and further classification of the information about person.

Keywords Infrared thermography (IT) · Artificial neural network facial identification · Support vector machine

1 Introduction

In today's age of automation, any detail is interpreted by an artificial intelligence computer and used in a variety of sophisticated applications. While several organizations have built state-of-the-art surveillance programs, recent terrorist attacks

K. Rani (✉) · M. Kalra

Department of Computer Science Engineering, National Institute of Technical Teachers Training and Research, Chandigarh 160019, India

M. Kalra

e-mail: malakalra@nitttrchd.ac.in

R. Kumar

Department of Computer Science Engineering, Central University of Haryana, Mahendergarh 123031, India

have uncovered significant vulnerabilities in advanced security systems. There have already been different stages in the development of face expression, particularly geometric methods involving the depiction of the facial in terms of lengths, positions, and areas among attributes such as pupils, nose, or chin [1–3]. Some of the researches have used Gabor filters to locations on the facial recognition scheme, but one tested and produced positive results versus geometric techniques. [4]. To overcome these issues and enhance quality of facial expression recognition the non-invasive methods like IT comes into picture. IT is widely accepted Non-Destructive Testing & Evaluation (NDT&E) approach for computer vision applications. It is the method of region inspection which is non-destructive, non-contact, quick, and comprehensive. IT measures the emission of radiant heat across the sample element and monitors fluctuations in temperature and relative humidity using thermal imaging camera. IT has a wide range of applications including defense [5], electrical [6], automotive [7], geological [8], agricultural [9], mechanical [10], aviation [11], and medical [12]. IT is roughly divided into two categories the passive and active approach to thermography, taking the measuring methods into account. The passive thermal imaging approach requires no external heating elements, whereas the active thermal imaging includes an external heating element such as physical, hydraulic, gravitational, or other means of excitation to improve the thermal contrast. IT and image processing together create a powerful approach in which machine learning algorithms have put their efforts to make an autonomous system for industrial application [13]. Significant improvements have been made in the face recognition issue by the work performed in [14]. However, on the basis of existing literature, no work has been conducted to solve the face recognition issues with thermal images. However, thermal images lose essential texture detail, particularly over the eyes and mouth, depending more on the body temperature of the person.

Feature-based face identifications [15] illustrated the potential of differences induced by illustration and achieved higher precision speeds. In order to create the mechanism of identifying illumination variation, the phase congruence features maps are used as inputs to the facial recognition system instead of intensity values. The selection method described in this paper is extracted from the definition of modular spaces [16]. Recognition methods focused on geographic areas have reached a high degree of precision. While facial images are influenced by variations such as non-uniform lighting and partial occlusions are still varied to local face regions. Modified images help to identify these differences, assuming that the modules produced are small enough. In this method, however, a large number of dependencies between different adjacent pixels could be overlooked. This could be overcome by rendering the units bigger, but this will result in an inaccurate placement of facial expression. In manner to resolve this issue, a technique for the development of modules has been applied in this article.

Considering multiple pixel dependencies around multiple sub-regions, this serves to offer additional details that could serve to increase the consistency of the classification. Also, the subspace methods such as PCA would not able to make the relationship among higher than two variables. The performance of the detection produced by thermal images is equal to that of visual images, using the methodology of feature

selection. This paper utilizes a judgement-level merger process. In the event of data level compression operations, the absolute best attributes (e.g., DWT coefficients) are chosen and the image is restored using these features.

2 Infrared Thermography-Based Face Recognition

This paper introduces a facial recognition model from infrared thermal images. The suggested model achieves an image description based on profound learning. The proposed methodology has five phases (i) thermal image collection (ii) preprocessing of raw thermal image (iii) extracting the regions of interest (iv) feature extraction and feature selection (v) finally the optimal features are given to SVM and ANN to test the recognition performance of system. The flow chart of proposed method is illustrated in Fig. 1.

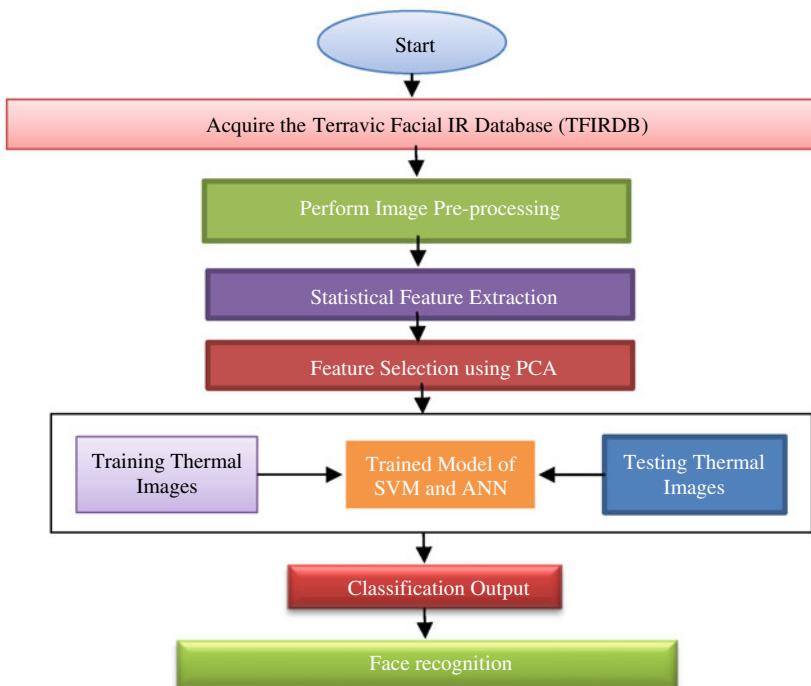


Fig. 1 Flowchart of methodology

2.1 Discrete Wavelet Transform (DWT)

DWT was one of the better decomposition methods and provides outstanding results. It is capable of conducting multi-resolution in both the frequency and time domain. DWT separates data into a variety of frequency components and analyzes each component more resolution compared to its size. The two-dimensional DWT decomposes an image at each step producing four sub-band images consisting of one approximation and three information coefficients as shown in Fig. 2. This method is approached independently in both ways, i.e., row and column. First, to really get the low and high-frequency constituents from rows and columns, high-pass and low-pass filters were deployed with each data repository and then down-sampled by 2. Both filterings are then added to the column, and then down-sampled by 2. Subsequently, four sub-band images are replicated: following main coefficients (HL, LH, and HH) and a low-resolution sub-image (LL) (Fig. 3).

Through sub-band picture has its own characteristic, as high-frequency components occur in detail coefficients as HH, LH, and HL bands. This same low-frequency portion is present in LL band that only decomposed in a comparable pattern during the next stage of sub-band signal decomposition. The range of the thermal image (I_0) at $k = 0$ can be particular by $2^{k=20}$. Moreover, the sub-part of images in the output result at $n = 1$ can be computed as

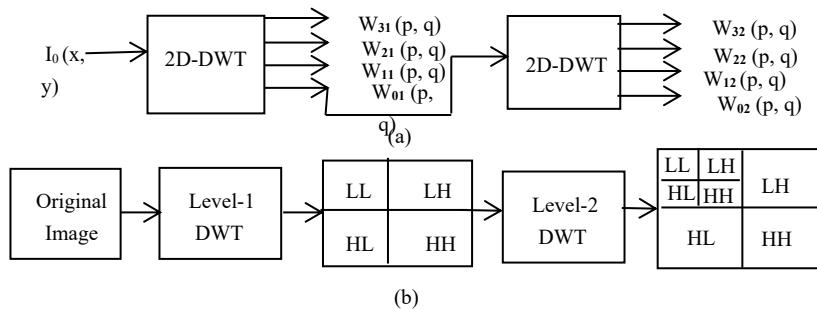


Fig. 2 Two-level 2D DWT **a** Sub-bands image **b** Wavelet sub-bands images

Fig. 3 Database Samples of Terravic Facial IR images



$$w_{01}(p, q) = [L_x * [L_y * I_o] \downarrow_2] \downarrow_2 (p, q) \quad (1)$$

$$w_{11}(p, q) = [L_x * [H_y * I_o] \downarrow_2] \downarrow_2 (p, q) \quad (2)$$

$$w_{21}(p, q) = [H_x * [L_y * I_o] \downarrow_2] \downarrow_2 (p, q) \quad (3)$$

$$w_{31}(p, q) = [H_x * [H_y * I_o] \downarrow_2] \downarrow_2 (p, q) \quad (4)$$

where the convolution then down sampling is represented by $*$ and \downarrow , respecs. Here (H_x, H_y) and (L_x, L_y) are high- and low-pass filters and down sampling and filtering steps for w_{01} may be given as

$$Y_{low}(p, q) = [L_y * I_o] \downarrow_2 (p, q) = \sum_{k=-2}^1 I_o(p, k) L_y(p, 2q - k), \quad (5)$$

$$w_{01}(p, q) = [L_x * [L_y * I_o] \downarrow_2] \downarrow_2 (p, q) = \sum_{k=-2}^1 Y_{low}(k, q) L_x(2p - k, q), \quad (6)$$

Similar process is taken to get w_{11} , w_{21} and w_{31} .

3 Feature Extraction

A variety of different approaches has been implemented and extract the image function. These features might include information about pixels, region, boundaries, and texture. Nonetheless, before receiving features, different images are applied to the sampled thermal image by preprocessing techniques such as thresholding, resizing, and normalization, binarization. Furthermore, the feature extraction was applied in order to obtain the features that will aid in the classification and diagnosis of thermal images. Here, the mean, standard deviation, variance, skewness, kurtosis, and root mean square have been used for the feature extraction. Finally, the extracted feature vectors are then correctly standardized to have zero mean and unity standard deviation to minimize the influence of dc offset and amplitude biases ranged from [0, 1] to the classifier.

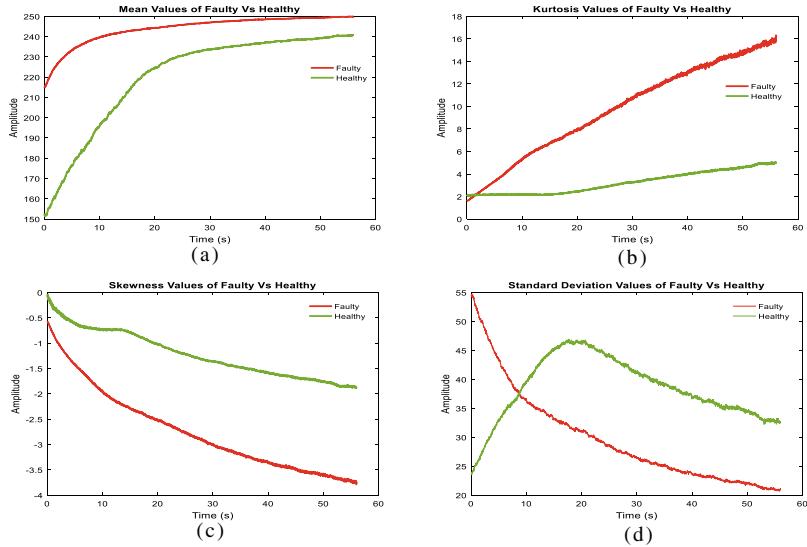


Fig. 4 Most relevant extracted features versus time plots for facial data **a** mean, **b** kurtosis, **c** skewness, and **d** standard deviation

4 Feature Reduction and Selection

The features extracted from the region of interest are suitable features, some may be unrelated to the fault, some may be related to each other, and redundancy could occur. PCA carries out the dimension reduction to reduce linear dependency between features. Because the healthy and faulty machines in the feature space are not linearly separable, more features need to be kept to improve the results of classification. PCA can only eliminate the linear relationship between features and is a powerful tool for extracting features or selecting features by keeping the data at maximum variance.

In manner to illustrate the characteristics of the extracted features different facial samples with respect to time are plotted. The most relevant feature vectors (i.e., mean, SD, kurtosis, and skewness) vs time plots are illustrated in Fig. 4. It can be observed among the feature vectors vs time plots that there is an inconsistency in different features and also show the noticeable changes in the amplitude of features with increase in time.

5 Results and Discussion

The first dataset has a sample, these have 3600 items; each thermal image is 320×240 pixels. These image pixels are split into a 76800-dimension of row vector, giving us 3600×76800 matrix X where each row is a facial images training instance. SVMs

and ANN are used as classifiers in this study because they give good properties as compared to other models, and with a wide training dataset, they may also provide higher classification accuracy. Nevertheless, when SVM is used for classification, an appropriate base kernel feature needs to be chosen for the SVM. Here, the feature vector dataset has been trained and testified for the classification by various SVM classifiers with different basis kernel function (i.e., linear, quadratic, cubic, fine Gaussian, medium, and coarse Gaussian) and the resultant overall classification accuracy has been analyzed. Throughout ANN, that gradient onto sigmoid is indeed a basic issue in neural networks and has been caused when our establishing a framework has high dimensions that will revert to all outputs after multiplication with both the weight matrix and implementing the sigmoid activation function whether 1 or 0, if that was the case, the local gradient onto sigmoid would be 0 ('vanish') from both situations. So it will get caught with dump neurons which not able to take the benefit from the developed network and decrease cost gradually. It is essential to run in a healthier zone called the effective sigmoid field.

For the improvement of the classification accuracy, the cross validation training and testing model has been employed to avoid overfitting and under fitting and to obtain optimize critical parameters of SVM. Each category is applied to test the trained model, while other categories are being used to train the model. The overall average accuracy of the classifier is an output result, and selected the different base kernel functions of SVMs to the output result with the supreme accuracy to build the final classification model. The classifier with the optimum features has been trained with whole normalized feature vector dataset followed by PCA using different n-fold (5-fold, 10-fold, 15-fold, 20-fold, and 25-fold) cross validation and aforementioned kernel function of SVMs for evaluating the classification rate.

Table 1 displays the identification performance of each classifier on Terravic facial IR dataset. Two main points can indeed be observed, the first one and more data we used during the training dataset, the maximum accuracy is achieved, which indicates that if we are using a massive database, we would get better performance, the second one the ANN is outer perform SVM due to even without the extraction feature, it could be classified with 100% overall accuracy than we use the ANN with

Table 1 Performance of the various classifiers on Terravic facial IR dataset

Classifier	Training set	Test set	Accuracy	Precision	Recall	F-Score
SVM	20	80	97.66	96.7	95.2	96.11
	40	60	99.5	98.1	97.45	97.26
	60	40	99.23	98.6	97.95	98.12
	80	20	99.87	99.1	98.27	98.59
ANN	20	80	98.02	98.01	97.5	97.6
	40	60	99.00	98.89	98.25	98.65
	60	40	99.55	99.01	99	98.80
	80	20	100	100	100	100

backpropagation method. Only 80 percent as a training data, while the best accuracy for the SVM and ANN method are 99.87 and 100 respectively percent as discussed in Figs. 5 and 6.

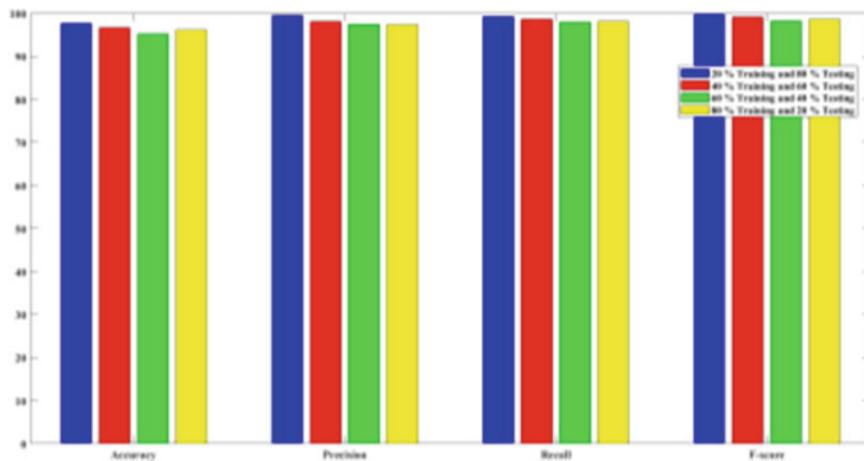


Fig. 5 Performance measure of SVM of the proposed work

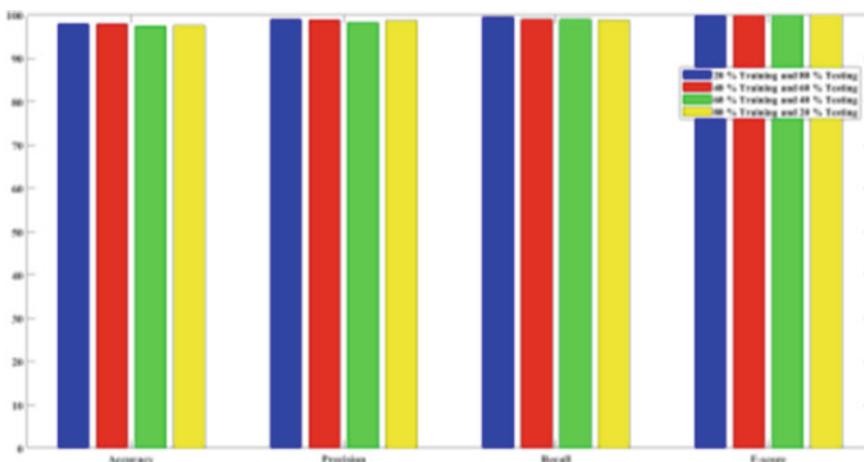


Fig. 6 Performance measure of ANN of the proposed work

6 Conclusion

Infrared facial identification is a widely used research field, but there is a lack of experiments in this area relative to visible facial recognition. In this article, the creation of these benchmarks is as important as the algorithms. Evolution. By proposing more and more complex datasets, they endorse and promote the development of new, better-performing techniques. Implementation of backpropagation has one secret layer, we were capable to correctly identify all the aspects of the thermal images and Terravic facial dataset excluding any functions. The SVM is capable of being graded with higher accuracy and precision on Terravic facial databases. But ANN is outer perform the SVM with 100% of accuracy. Compared with the conventional infrared image recognition approaches that mostly involve the basic process: face preprocessing is done and attributes are removed, qualitative decrement is used to minimize the dimensionality of the file, afterthought classification model is trained to obtained the performance. Out of the test dataset, ANN with backpropagation can identify from test data set with 100% accuracy using 80% of training data few results. The performance of the various algorithms is reasonable, but we need to decrease the length of the training process phase.

References

1. F. Prokoski, History, current status, and future of infrared identification, in *Proceedings of The IEEE Workshop on Computer Vision Beyond the Visible Spectrum (CVBVS'00)*, (Hilton Head, SC, 2000)
2. S. Kong, J. Heo, B. Abidi, J. Paik, M. Abidi, Recent advances in visual and infrared face recognition: a review. *Comput. Vis. Image Underst.* **97**, 103–135 (2005)
3. X. Chen, P. Flynn, K. Bowyer, IR and visible light face recognition. *Comput. Vis. Image Underst.* **99**, 332–358 (2005)
4. M. Akhloufi, A. Bendada, J.C. Batsale, State of the art in infrared face recognition". *Quant Infrared Thermogr J (QIRT)* **5**(1), 3–26 (2008)
5. A. Choudhary, S.L. Shimi, A. Akula, Bearing fault diagnosis of induction motor using thermal imaging, in *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, (Greater Noida, Uttar Pradesh, India, 2018), pp. 950–955, <https://doi.org/10.1109/gucon.2018.8674889>
6. A. Choudhary, D. Goyal, S.S. Letha, Infrared thermography based fault diagnosis of induction motor bearings using machine learning. *IEEE Sensors J.* (2020)
7. R. Schulz, S. Verstockt, J. Vermeiren, M. Locufier, K. Stockman, S. Van Hoecke, Thermal imaging for monitoring rolling element bearings, in *12th International Conference on Quantitative Infrared Thermography* (Bordeaux, France, 2014), pp. 7–11
8. R. Vadivambal, D.S. Jayas, Applications of thermal imaging in agriculture and food industry—a review. *Food Bioprocess Technol.* **4**(2), 186–199 (2011)
9. R.J. Grasso, Defence and security applications of quantum cascade lasers, in *Optical Sensing, Imaging, and Photon Counting: Nanostructured Devices and Applications 2016*, vol. 9933, p. 99330F (International Society for Optics and Photonics, 2016)
10. M. Zhiguo, R. Zhao, Z. Cai, J. Ping, Z. Tang, S. Chen, Microwave thermal emission at Tycho area and its geological significance. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **10**(6), 2984–2990 (2017)

11. S.S. Alaviyoun, M. Ziabasharhagh, Experimental thermal survey of automotive turbocharger.. *Int. J. Engine Res.* **21**(5), 766–780 (2020)
12. A. Bandyopadhyay, A. Sengupta, A review of the concept, applications and implementation issues of terahertz spectral imaging technique. *IETE Techn. Rev.* **7**, 1–9 (2021)
13. R.S. Ghiass, O. Arandjelovic, H. Bendada, X. Mal dague, Infrared face recognition: a comprehensive review of methodologies and databases. *Pattern Recogn.* (2014)
14. K.R. Kakkirala, S.R. Chalamala, S.K. Jami, Thermal infrared face recognition: a review, in *UKSim-AMSS 19th International Conference on Computer Modelling & Simulation (UKSim)* (2017), pp 55–60
15. R. Rojas, *Neural Networks: A Systematic Introduction* (Springer-Verlag, Berlin, New-York, 1996)
16. V. Chudasama, K. Upla, RSRGAN: computationally efficient real-world single image super-resolution using generative adversarial network. *Mach. Vis. Appl.* **32**(1), 1–18 (2021)

An Efficient Cluster Assignment Algorithm for Scaling Support Vector Clustering



H. S. Jennath and S. Asharaf

Abstract Support Vector Clustering (SVC) algorithm reformulates SVM's Quadratic Programming as a minimum enclosing ball (MEB) problem, where every point in the data space will be projected to the higher dimensional feature space to find the minimum radius that encloses all the data points inside the sphere. Support vectors are data points in the surface of the minimum enclosing ball. The clustering algorithm works by mapping the support vectors of MEB back to the data plane forming groups or derives a contour that encloses a set of clustered points. However, the major limitation of this algorithm is that it fails to scale with larger dataset. Computation bottleneck lies in the efficient cluster computation approach and in solving the QP optimization for MEB. In order to solve this cluster computational complexity, this work presents a simple, efficient cluster assignment algorithm using similarity of feature set for data points in high-dimensional feature space utilizing an efficient MEB approximation algorithm. Experiments are carried out by varying the similarity threshold metrics. Performance of the clustering mechanism run on various datasets demonstrates the proposed algorithm is lighter, simple, and converges faster.

Keywords Support vectors · Core-sets · Cosine similarity · Core vector machine · Support vector clustering · Kernel machines · Rkhs

1 Introduction

Clustering is an unsupervised learning paradigm in machine learning that explores potentially hidden relationships or unidentified groups present in the dataset based on some similarity metrics. Various clustering approaches exist in the literature that

H. S. Jennath (✉)

CUSAT Research Center, Technopark Campus, Indian Institute of Information Technology and Management-Kerala, Trivandrum 695581, India

e-mail: Jennath.res16@iitmk.ac.in

S. Asharaf

Indian Institute of Information Technology and Management-Kerala, Technopark Campus, Technopark Campus, Trivandrum 695581, India

e-mail: Asharaf.s@iitmk.ac.in

makes use of parametric as well as non-parametric estimations. Non-parametric clustering algorithms like Support Vector Clustering(SVC) [1], core vector machine-based clustering [2, 3, 17], etc., that makes use of support vector approach of kernel machines [11, 12] for non-linear cluster estimation.

Support vector clustering algorithm is a non-linear cluster estimation algorithm that formulates the SVM's QP as a minimum enclosing ball (MEB) problem. Minimum enclosing ball (MEB) problem is defined as for any given set X , which contains m points such that $X = x_1, \dots, x_m$ is such that every point x_i in the data space will be projected to the higher dimensional feature space to find the minimum radius that encloses all data points inside the sphere. In the feature space, it searches for the smallest sphere, which we refer to as the minimum enclosing ball(MEB) that encloses the image of the data. The SVC clustering algorithm works by mapping the support vectors of MEB back to the data plane to form the contour boundary, which involves a lot of computation. In short, the computation bottleneck of SVC lies in solving the QP optimization for MEB and the cluster allocation approach for large datasets.

Sun et al. [27] proposed a SVC-based K-means clustering using Minimum Spanning Tree Pruning (MSTP) to initialize the number of clusters and clustering centroids. Asharaf et al [30] proposed a rough set theoretic approach of support vector clustering to design soft clustering. Chiang et al. [28] proposed a multi-sphere clustering algorithm based on adaptive cluster cell growing method using a new kernel-based fuzzy clustering approach. Yang et al. [29] proposed a robust cluster assignment method that uses proximity graphs to model the proximity structure of the data. However the major limitation of SVC is w.r.t scalability and complexity. Tsang, Ivor, et al. [2] proposed an efficient approximate MEB problem termed CVM. The advantage of CVM is that the complexity of it is linear with respect to m , while the space complexity is independent of m .

The Core Vector Machine (CVM) may be viewed as an efficient solution to the QP (quadratic programming) optimization problem and the proposed approach as an efficient solution to the cluster assignment problem. This work develops a simple threshold-based non-linear cluster estimation of support vectors in high-dimensional feature space using Gaussian kernels with lesser computation using a similarity distance measure. The proposed algorithm employs a better MEB approximation algorithm for resolving QP issue termed, core vector machine (CVM) designed by Tsang et al. [3] which is a combined architecture of SVM learning along with efficient approximate computational geometry of MEB estimation.

The CVM works on finding core-sets, which is an extract of a small set of points from the datasets, which internally represents the geometry of the data points. This clearly shows us core-sets enables an efficient building of approximate clustering [25]. In short, CVM is a $(1+\epsilon)$ approximation of MEB algorithm to obtain a near-optimal SVM solution. Samadian et al. [7] proposed an unconditional core-sets using regularization for loss minimisation. Building large labeled image dataset for training convolutional neural network is very expensive and hard to create. Sener et al. [4] have proposed active learning as a solution to core-set selection, to choose set of points such that a model a selected subset is a better representation for the remaining

data point. Sinha et al. [5] propose a core-set-based approach for speeding up the gan training. The proposed approach in this work is a core-set-based clustering algorithm using cosine similarity.

The proposed approach is based on the closeness measure of the data points concerning the anchor points in kernel space within MEB. The threshold is based on the closeness measure that decides the width and number of clusters. A stochastic leader algorithm (random selection) is used to select anchor points from the list of support vectors. For each anchor point, all available SVs and other data points are iterated and if any SV being considered to have the closeness distance within the threshold for the anchor point, then it will be added to the cluster and will be removed from the available list. This algorithm will be exhaustively run till all SVs are assigned. The contour thus created in the data space will be the cluster boundary. Any data point that falls outside the threshold of clusters needs to be included, nearest neighbor algorithm is run on the Internal data points, to get them mapped into available clusters. In a nutshell, this work proposes a non-parametric clustering algorithm that uses distance metrics that measures the similarity of the data points in Reproducible Kernel Hilbert Space (RKHS).

The rest of the paper is organized as follows. Section 2 reviews related literatures and Sect. 3 provides an introduction to the proposed approach. An evaluation of the proposed approach on various datasets and comparative results are discussed in Sect. 4. Section 5 concludes the work.

2 Background Literature

This section covers the literature related to the supporting algorithms used for the proposed work such as core vector machine and ball vector machine.

2.1 *Background: Core Vector Machines*

Core Vector Machines (CVM) employs an efficient approximation algorithm for solving QP optimization for the search of MEB in kernel space. The problem-solving approach is by transforming the QP to an MEB problem, and then employs an iterative $(1 + \epsilon)$ approximation algorithm to obtain an approximate-optimal solution. In short, the crux of the core vector machine algorithm is to maintain a core-set C_s , which is a subset of set S, in such a fashion that, its MEB is computed at each iteration t, $B_t(C_*, R(*))$. If any point from set S falls outside the MEB in the feature space, that point will be included in the core-set by expanding the MEB by $B(C, R(1 + \epsilon))$. This is repeated until all points are evaluated if it falls within the computed MEB. However, after exhausting all points in set S, $(1 + \epsilon)$ approximate solution of MEB, i.e., $B(C, R)$ will be obtained. From this step, the primal variables associated with SVM like bias, weights, slack errors, etc., could be deduced.

2.2 ***Background: Ball Vector Machine***

The CVM is tightly coupled to the MEB problem. In Ball Vector Machine, unlike finding the minimum enclosing ball in CVM, BVM solve for an enclosing ball (EB) for a given prior radius. The BVM algorithm is as given in Algorithm ???. The ball's center is updated such that the new ball just touches $\psi(z_t)$. The whole procedure is repeated until no point falls outside $B(c_{t+1}, (1 + \epsilon)r)$. This produces an $(1 + \epsilon)$ -approximation.

In short BVM defines an enclosing ball problem instead of MEB such that for given a radius $r \geq R^*$, it finds a ball $B(c, r)$ with center c and radius r that encloses all points in S , such that

$$\| c - \psi(z_i) \| \leq r^2 \quad \forall \psi(Z_i)$$

in space S . In BVM, they propose an $(1 + \epsilon)$ approximation algorithm for solving the EB problem in an efficient methodology for growing or large scale data with lesser complexity.

This approach is far better than the native CVM approach as the complexity of numerical optimization is avoided with the mere computation of C_{delta} instead of continually computing the MEB using convex optimization performed over smaller chunks of data. This computed delta is added to the previously computed ball center c_t in order to accommodate the datapoint that falls outside the enclosing ball.

3 Efficient Cluster Computation : Proposed Approach

The inherent limitation of the SVC is the non-linear computation complexity of cluster assignment for larger datasets. According to the SVC algorithm, for every pair of support vectors in the data plane, a geometric operation involving $F(x)$ is being carried out in feature space. For any pair of points that belong to different clusters in data space, the path connecting the SVs must exit from the MEB boundary in the feature space. Any pair of points that belong to the same boundary should have its path bounded within the MEB in the feature space. This computation is to be repeated for every pair of support vector in the dataset. Binary valued Adjacency matrix (AM) is to be built based on the above computation, with zeros if the path exited from the MEB sphere, or with ones if the pair of points belonging to the same cluster.

In short, this approach of transforming sample points from any path connecting support vectors in data space to the higher dimensional feature space, through kernel computation to identify the connected components is a complex computation approach. Say n is the number of support vectors, m is the no of samples taken for the line joining any two support vectors, o is the computation time required to check if the point is inside MEB or not, then the computation time required for cluster



(a) Native Support Vector Clustering. (b) Proposed algorithm over QP Opt

Fig. 1 Clustering using QP optimisation: SVC versus Proposed approach

formation will be of the order of $nC2^*o^*m$. The existing cluster formation based on projecting the points joining the SVs to the RKHS plane to check to the two SVs fall within a cluster is computationally intensive for large datasets. Moreover, this approach fails to scale with larger datasets.

3.1 Proposed Efficient Clustering Approach

In this work, we propose a simple scalable clustering algorithm based on the similarity threshold of SVs in high-dimensional feature space using non-linear kernel machines. A leader algorithm is used to identify the anchor points. For every anchor point, connected components are identified from the similarity boundary, limited by the given threshold. The proposed clustering algorithm address the complexity of the cluster assignment. This algorithm is first executed over the support vectors obtained from QP optimization of MEB in RKHS. The performance of the proposed clustering approach is compared against the native SVC clustering.

Schematic of the 2d clustering of the proposed algorithm versus SVC for iris dataset is depicted in Fig. 1.

Moreover, the SVC algorithm has the inherent limitation of non-linear increase in complexity of computation of MEB concerning the volume of datasets. To overcome the limitation of QP optimization of finding MEB, we chose a Core vector machine (CVM) that offers a better approximation algorithm for finding MEB in RKHS.

The interesting aspect of the core-sets is that their size is independent of the dimensions. For larger datasets, the CVM has empirically proved to have better classification performance in [2, 17] and is good in regression in [2] and semi-supervised learning termed BVM [3]. Also, this relationship between SVM and MEB has employed in various works such as Nock et al. [22] one-class classification with arbitrary Bregman divergence, MEB for support vector ordinal regression [23] etc. However, CVM has not explored the direction of clustering using the cosine similarity of the feature sets in RKHS.

The core-sets obtained from the CVM is the approximation of the MEB which in turn is the approximation of the marginal SVs (support vectors). The complex and computationally heavy single convex optimization of all the data points to compute the MEB in SVC is replaced with a simple CVM approach to obtain the approximate

MEB. The core-sets could be derived from the scalable CVM algorithm, which is processed as a batch of chunks iteratively at any predefined sampling rate, offers a viable solution for larger datasets. Hence CVM is offering a scalable approach for obtaining the support vectors which best describe the dataset.

This is the input for the cluster assignment algorithm. All points whose cosine similarity is within the predefined threshold with respect to the anchor SV are added to the cluster, formed by the anchor SV. The remaining data points if any not in any cluster could be allocated to the clusters using the nearest neighbor approach. The proposed clustering approach using the CVM approximation is given below in Algorithm 1.

Algorithm 1: Proposed Algorithm using Core Vector Machine for Clustering

Result: Clusters with respective anchor points

Input1: Get CoreSets and its weights from CVM

Input2: Data points Dp

Initialization: $K_{i,j}$ = Compute the cosine similarity measure between the Coresets and Data points

$S_t h$ = Given threshold

C_s = No of elements in Coreset

$Set(C_s)$ = Coreset from CVM

D_p = Datapoints to be clustered

C_i = Cluster with anchor point as i

while $i \in Set(C_s)$ **do**

$C_s = C_s - 1$

 Form cluster with i as Anchor node

while $j \in (D_s)$ **do**

if $K_{i,j}$ falls within $S_t h$ **then**

C_i = Add j as i's cluster member

 Remove j from the set Ds and Cs(if present)

else

 | continue;

end

end

if $len(D_s) \geq 0$ **then**

 Compute mean centre for identified cluster

while $j \in (D_s)$ **do**

 1. Perform Nearest Neighbour assignment

 2. Remove from Ds list

end

else

 | Break;

end

end

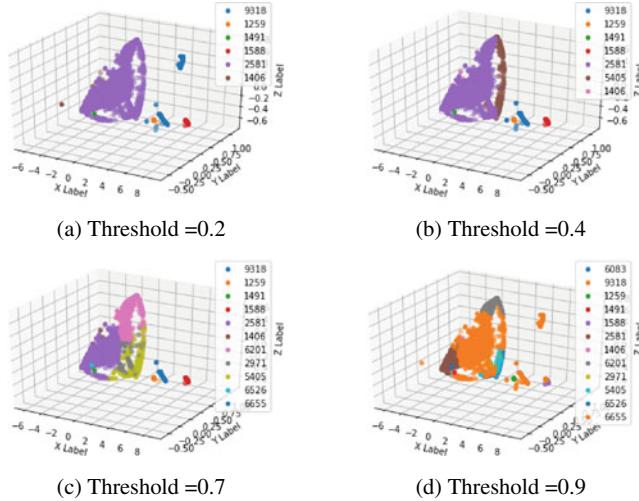


Fig. 2 Proposed algorithm with CVM: 10K :KDD Dataset

However, the improved CVM termed BVM (ball vector machine) offers an efficient ball estimate which requires only $\mathcal{O}(\frac{1}{\epsilon^2})$ iterations to achieve similar/better performance of CVM. Hence for the experimentation, we used the improved CVM, termed BVM for extracting core-sets. To mitigate confusion, the notation used is CVM for experimental evaluations. Figure 2 refers to the three-dimensional view of clustering based on various similarity thresholds for 10K data points from the KDD dataset. The principal component analysis is performed to determine the three important dimensions or features to be plotted in the graph.

In a nutshell, the proposed clustering approach iterates the support vectors or core vectors across the set of input vectors. This is done to compute the cosine similarity of data vectors concerning the support vectors in RKHS. Cluster formation is limited by the given threshold of the similarity, in high-dimensional space. Hence it takes computations of the order $\max(m \cdot n)$ for cluster computations, where m is the number of support vectors and n is the number of internal data points. As m is very less than n for CVM, the computation complexity of cluster assignment in the proposed paper will be more or less linear to the size of the dataset.

3.2 Experimental Setup

Experiments are performed on 2 datasets, viz., iris dataset and the KDD dataset [26]. The proposed algorithm employed the similarity of the data points featured in the high-dimensional kernel space for clustering. The cosine value computed in the RKHS space using the Gaussian kernel, between support vectors and other data

Table 1 Parameters used for QP optimisation

Parameter names	Parameter values
C	0.08
q	0.5
Cut value	10**-8

points gave the measure of similarity of those two points concerning their feature set. The proposed clustering algorithm is run over QP optimization using the wolf dual method to compare the performance of SVC clustering. To address the MEB limitation, the proposed clustering is combined with an approximate MEB algorithm (CVM) to discover clusters based on their closeness in RKHS. The purpose of these (SVC's QP optimization and CVM's approximate MEB optimization) underlying algorithm was to derive the support vectors from the provided datasets.

For this experimentation, SVM parameters are set as given in Table 1. C parameter is the regularization param set as 0.0800, kernel parameter q is set to 0.5, cut off value of beta co: coefficients of QP optimization to determine SVs and Border support vectors is set to 10**-8. All experiments are conducted on an Intel i5 CPU with 1.8 GHz clock speed with 12 GB ram in a Linux Machine.

4 Results and Discussions

The proposed clustering algorithm run on the iris dataset with 150 data points using QP optimization took 19 s whereas the classical Support vector clustering approach took 162 s for clustering. QP optimization took around 14 s for 150 data points. Table 2 corresponds to the performance of the proposed clustering approach with QP optimization using iris dataset vs SVC approach. This is because the computation time in the proposed approach is linear, $\mathcal{O}(n)$ w.r.t to size of the dataset. With respect to increase in data points, the processing time is increasing in exponential order for the classical SVC approach. However by increasing data points to an order of a few hundred, the clustering algorithm of naive SVC fails miserably. Performance of the proposed cluster assignment approach with SVC's QP optimization and native SVC clustering is as shown in Fig. 2. Moreover, this approach fails to scale with larger datasets.

The proposed algorithm with sorted support vectors based on the optimization weights for cluster assignment is shown in Fig. 3. Figure 4 shows a random selection of SVs for cluster assignment with a varying threshold of similarity distance across the neighboring data points.

Table 2 Performance of proposed algorithm using QP optimisation vs SVC using Iris dataset

Dataset Size	QP Optimisation time in seconds	SVC computation time in seconds	Computation time of proposed algorithm
50	14	26	16
75	14.2	39	16.8
100	14.3	62	17.2
120	14.8	95	17.6
150	14.9	162	19

Table 3 Parameter Values used in experimentation

Parameter for CVM	Parameter Values
gamma	0.01
epsilon	0.1
C	10

Table 4 Performance of the proposed algorithm using CVM: KDD Dataset

Data size	QP Optimisation time in seconds	CVM Optimisation time in seconds	Proposed Clustering Algorithm time in seconds	Total running time for proposed algorithm using CVM
1000	65	62.14	5.3	67.44
1500	133	66	6.15	72.15
3000	537	72.13	7.12	79.25
10000	More than 2 hrs	78.82	14.25	93.07
30000	More than 2 hrs	79.15	20.34	99.49

Increasing the number of data points to few thousand tuples, say around 2000, QP optimization stoops its performance as shown in Table 4. QP is taking around 500 s for MEB estimation of three thousand points. Increasing the size of the dataset beyond this point fails QP optimization miserably.

CVM offers comparable performance in determining the core-sets of larger datasets, by using the MEB or EB ($1+\epsilon$ approximation algorithms, respectively. A Gaussian kernel is used with parameters such as gamma, epsilon, and regularization parameter C is as shown in Table 3.

Computation time for core-sets for datasets of size 3000, 10000, 30000, etc., is not varying too much. CVM is approximately taking around 60–70 s for deriving the core-set for data size ranging from 3 to 30 K as shown in Table 4. Also, the size of the core-set is independent of the size of the dataset. The computation time of the proposed cluster-based similarity distance measure algorithm is shown in Table 4. The computation time of the Native SVC approach is not shown in the

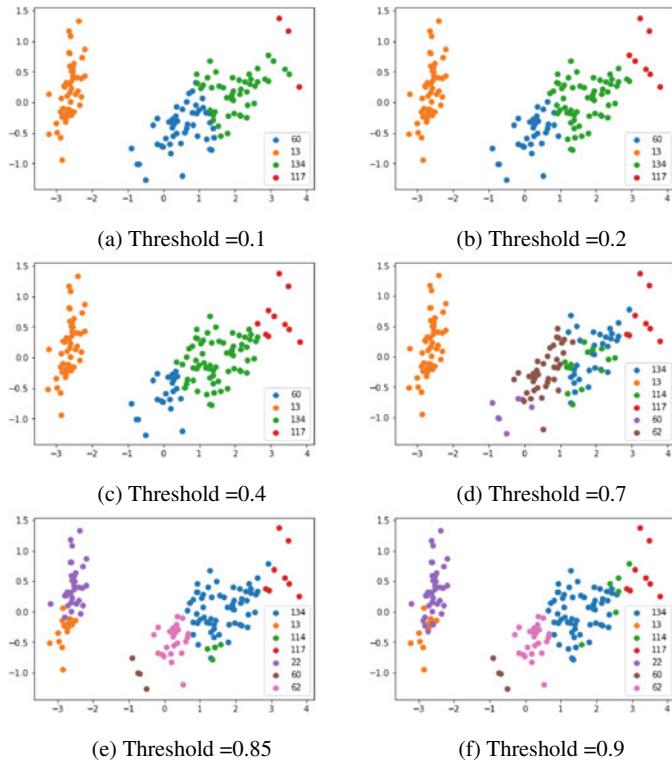


Fig. 3 Proposed algorithm with sorted support vectors-SVC: Iris Dataset

graph as it is taking computation time of the order of 100s of minutes which is beyond the comparative limit and hence not included in the table. Figure 2 depicts the 3D schematic of cluster allocation using the proposed approach using the varying value of similarity threshold in RKHS. Three dimensions for plotting the diagram are chosen using Principal Component Analysis.

5 Conclusions

To address the limitation of the SVC clustering approach, in this work, we developed a lite, simple, efficient cluster assignment algorithm using a distance measure of similarity between data points in high-dimensional feature space. The complexity of the native SVC algorithm limits its performance when executed on large datasets. Computation overheads include solving the QP optimization for finding MEB in high-dimensional feature space and computation complexity in cluster assignment approach for larger datasets. To overcome the limitation of MEB computation for a

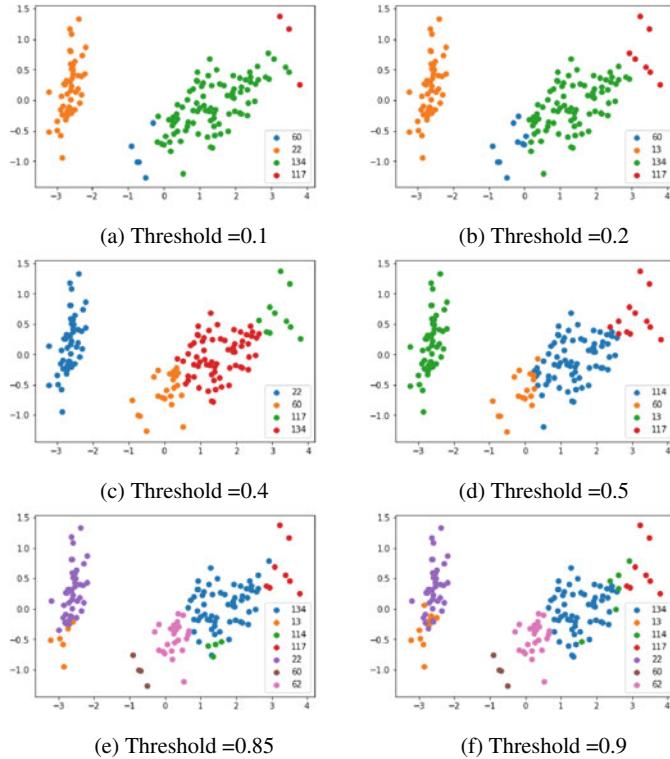


Fig. 4 Proposed algorithm with random support vectors-SVC: Iris Dataset

larger dataset, we employed an efficient approximate MEB computation algorithm termed CVM. For the cluster assignment, a proximity-based algorithm was developed. Clustering is formed by iterating the support vectors and the data points based on a threshold of similarity. For a given threshold for similarity, elected anchor points form clusters, based on the degree of familiarizing it holds with the neighboring points in the feature set using dot product in RKHS. Experimental results show that the proposed algorithm has tremendous performance compared to existing SVC-based implementations, also it can handle very large datasets, and is even faster than native SVM-based approaches.

Acknowledgements This work is supported by the Back to Lab Program research fellowship (Ref: Order No.1281/2016/KSCSTE) from Women Scientists Division (WSD), Kerala State Council for Science, Technology, and Environment (KSCSTE).

References

1. A. Ben-Hur, D. Horn, H.T. Siegelmann, V. Vapnik, Support vector clustering. *J. Mach. Learn. Res.* **2**(Dec), 125–137 (2001)
2. I.W. Tsang, J.T. Kwok, P.M. Cheung, Core vector machines: fast SVM training on very large data sets. *J. Mach. Learn. Res.* **6**(Apr), 363–392 (2005)
3. I.W. Tsang, A. Kocsor, J.T. Kwok, Simpler core vector machines with enclosing balls, in *Proceedings of the 24th International Conference on Machine learning* (2007), pp. 911–918
4. O. Sener, S. Savarese, Active learning for convolutional neural networks: a core-set approach (2017). arXiv preprint [arXiv:1708.00489](https://arxiv.org/abs/1708.00489)
5. S. Sinha, H. Zhang, A. Goyal, Y. Bengio, H. Larochelle, A. Odena, Small-gan: speeding up gan training using core-sets, in *International Conference on Machine Learning* (PMLR, 2020), pp. 9005–9015
6. O. Bachem, M. Lucic, A. Krause, Practical coresets constructions for machine learning (2017). arXiv preprint [arXiv:1703.06476](https://arxiv.org/abs/1703.06476)
7. A. Samadian, K. Pruhs, B. Moseley, S. Im, R. Curtin, Unconditional coresets for regularized loss minimization, in *International Conference on Artificial Intelligence and Statistics* (PMLR, 2020), pp. 482–492
8. K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl, Constrained k-means clustering with background knowledge. *Icml* **1**, 577–584 (2001)
9. M. Ester, H.P. Kriegel, J. Sander, X. Xiaowei, A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd* **96**(34), 226–231 (1996)
10. M. Ankerst, M.M. Breunig, H.P. Kriegel, J. Sander, OPTICS: ordering points to identify the clustering structure. *ACM Sigmod Rec.* **28**(2), 49–60 (1999)
11. K.R. Muller, S. Mika, G. Ratsch, K. Tsuda, B. Schölkopf, An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Netw.* **12**(2), 181–201 (2001)
12. B. Schölkopf, A.J. Smola, F. Bach, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond* (MIT press, 2002)
13. V. Vapnik, *Statistical Learning Theory* (Wiley, New York, 1998)
14. B. Schölkopf, A. Smola, K. Müller, Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **10**(5), 1299–1319 (1998)
15. F. Murtagh, A survey of recent advances in hierarchical clustering algorithms. *Comput. J.* **26**(4), 354–359 (1983)
16. R. Yager, Intelligent control of the hierarchical agglomerative clustering process. *IEEE Trans. Syst. Man Cybern.* **30**(6), 835–845 (2000)
17. S. Asharaf, M.N. Murty, S.K. Shevade, Cluster based core vector machine, in *Sixth International Conference on Data Mining (ICDM'06)* (IEEE, 2006) (pp. 1038–1042)
18. J. Cervantes, X. Li, Y. Wen, K. Li, Support vector machine classification for large data sets via minimum enclosing ball clustering. *Neurocomputing* **71**(4–6), 611–619 (2008)
19. P. Kumar, J.S.B. Mitchell, E. Alper Yildirim, Approximate minimum enclosing balls in high dimensions using core-sets. *J. Exp. Algorithmics (JEA)* **8**, 1–1 (2003)
20. S.D. Ahipasaoglu, E.A. Yildirim, Identification and elimination of interior points for the minimum enclosing ball problem. *SIAM J. Optim.* **19**(3), 1392–1396 (2008)
21. Y. Wang, Y. Zou, S. Zheng, X. Guo, Simpler minimum enclosing ball: fast approximate MEB algorithm for extensive kernel methods, in *2008 Chinese Control and Decision Conference* (IEEE, 2008), pp. 3576–3581
22. R. Nock, F. Nielsen, Fitting the Smallest Enclosing Bregman Ball, in *European Conference on Machine Learning* (Springer, Berlin, Heidelberg, 2005)
23. S.K. Shevade, W. Chu, Minimum enclosing spheres formulations for support vector ordinal regression, in *Sixth International Conference on Data Mining (ICDM'06)* (IEEE, 2006), pp. 1054–1058
24. S. Har-Peled, D. Roth, D. Zimak, Maximum margin coresets for active and noise tolerant learning, in *IJCAI*, pp. 836–841

25. M. Badoiu, S. Har-Peled, P. Indyk, Approximate clustering via core-sets, in *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing* (2002), pp. 250–257
26. KDD Cup (1999), <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>. Accessed Oct 2007
27. Y. Sun et al. A novel SVC method based on K-means, in *2008 Second International Conference on Future Generation Communication and Networking*, Vol. 3 (IEEE, 2008)
28. J.H. Chiang, P.Y. Hao, A new kernel-based fuzzy clustering approach: support vector clustering with cell growing. *IEEE Trans. Fuzzy Syst.* **11**(4), 518–527 (2003)
29. J. Yang, V. Estivill-Castro, S.K. Chalup, Support vector clustering through proximity graph modelling, in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02*. Vol. 2 (IEEE, 2002)
30. S. Asharaf, S.K. Shevade, M. Narasimha Murty, Rough support vector clustering. *Pattern Recogn.* **38**(10), 1779–1783 (2005)

In Silico Analysis of Plant-Derived Medicinal Compounds Against Spike Protein of SARS-CoV-2 and Ace2



Tanya Sharma, Mohammad Nawaiid Zaman, Shazia Rashid,
and Seneha Santoshi

Abstract Introduction: Even after many months, the highly contagious and infectious SARS-CoV-2 virus that caused a pandemic in the year 2020 is still thriving. The identification of the effective drug against this virus is the need of the hour. This highly infectious virus, upon entry into the host, binds to the cellular Angiotensin-converting enzyme 2 (ACE2) with the aid of spike protein. Thus, both of these proteins represent a potential therapeutic target for inhibiting the interaction between the two. This study utilizes the *in silico* approach to identify compounds that can inhibit the interaction of the spike protein of the SARS-CoV-2 and ACE2. The study investigated plant-derived medicinal compounds and the two compounds and their derivatives (Arjunolic acid, Noscapine) based on their anti-viral properties. **Methodology and tools used:** Using molecular docking techniques, the binding affinity of the compounds with the protein targets were measured, and then the results were analyzed by building the interaction plots and mapping the binding residues on the interface of the interaction. The methodology used was carried out using AUTODOCK Vina 1.1.2, PyMOL, and LigPlot Plus v 2.2. **Results:** The results indicate that out of 40 compounds used in this study, Withanolide, Arjunolic Acid, Calceolarioside, and Amaranthin Beta-cyanin had a high affinity for the spike protein, ACE2 and the molecular complex of both (spike protein of SARS-CoV-2 and ACE2). **Conclusion:** In conclusion, all these identified compounds have the potential to disrupt the binding of the spike protein and ACE2 as well as carry an additional advantage of imparting relief in symptoms and may be investigated for further research to find a cure.

Keywords SARS-CoV-2 · ACE2 · Molecular docking · Receptor binding domain · Antivirals · Medicinal plants · Arjunolic acid · Noscapine · AutoDock · Spike protein

T. Sharma · M. N. Zaman · S. Rashid · S. Santoshi (✉)
Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida 201 313, India
e-mail: ssantoshi@amity.edu

S. Rashid
e-mail: srashid@amity.edu

1 Introduction

Early January 2020, the World Health Organization (WHO) announced the SARS-CoV-2 outbreaks as a Public Health Emergency. WHO characterized it as a pandemic as the infection spread in more than 18 countries worldwide [1]. Till now, more than 42 million people have been infected with SARS-CoV-2 and more than 1.6 million deaths have been caused worldwide due to SARS-CoV-2 [2].

Coronaviruses are single-stranded positive-sense RNA viruses with large viral RNA genomes [3]. It is highly infectious, contagious, and gives rise to severe pneumonia-like respiratory symptoms with a dry cough, fever, headache and difficulty in breathing, tiredness, chest pain and pressure, and loss of taste and smell [4]. The spike protein of the SARS-CoV-2 comprises S1 and S2 domains where S1 is known as the Receptor Binding Domain(RBD) [5].

Angiotensin-converting enzyme 2(ACE2) present on the epithelium of the nose, mouth, and membranes of many cell types such as lungs, arteries, heart, and intestines in humans. In humans, it performs many roles and is crucial for regulating processes like blood pressure, wound healing, and inflammation [6]. Many reports have identified that SARS-CoV-2 has the highest binding affinity for human ACE2 [7]

SARS-CoV-2 enters the host cells when the host comes in close proximity with the infected person or when a person inhales the small droplets produced by an infected person [8]. Upon entry, the SARS-CoV-2 interacts with the ACE2 via the RBD of the spike protein anchored on the outer covering of the virus [9]. The binding of RBD and ACE2 is facilitated by the interaction residues present in RBD (Leu455, Phe486, Asn487, Asn501) and the residues on the ACE2 receptor(Lys31, Glu35, Asp38, Lys353) [10]. SARS-CoV responsible for causing SERS by interacting with the ACE2 via two hotspot residues [11]. These two hotspots of amino acid in the ACE2 are called hotspot 31 and hotspot 353 are made up of salt bridges between Glu35-Lys 31 and Asp38-Lys353 on the ACE2 which are responsible for the binding of receptor and RBD spike proteins via favorable interactions [11, 12]. It has been suggested that there is 70% similarity between SARS-CoV-2 and SARS-CoV due to which SARS-CoV-2 recognizes Lys31 hotspot residue via Gln493 and Leu 455 [13]. This interaction is supported by other residues as well.

In view of the above, it has become urgent and crucial to identify all the interactions possible among the host and the virus protein to screen for a potential inhibitor that disrupts binding between the host and the SARS-CoV-2 viral proteins and can prove valuable for drug design. This present study is focused on screening the potential candidates which have an affinity for either both the targets in a complex form, the spike protein present on the envelope of the SARS-CoV-2 and the ACE2 Receptor in humans, or any one of them so that they can further be explored in the context of drug design. In this study, potential compounds were identified using in silico methods that can interact with the above-mentioned in order to disrupt the contact between the virus and the host. For several decades, plant-derived medicines have been in use to treat different diseases and disorders [14]. Plants are considered to be one of the main

sources of biologically active products that have fewer side effects and have shown antioxidant, antimutagenic, antiviral, and anti-inflammatory properties [15]. The aim was to identify compounds from medicinal plants that can inhibit the spike protein or ACE2 receptor either by binding directly to the contact residues or indirectly by binding to the surrounding residues that might lead to a change in the conformation of the active site which may inhibit the interaction of spike of SARS-CoV-2 and ACE2 receptor. In this study, a thorough literature search was carried out to identify compounds from medicinal plants that exhibited the potential antiviral properties and were screened to study their interactions with the spike protein or ACE2 receptor [16]. Along with the compounds which are known to have therapeutic properties in respiratory diseases, two more compounds and their derivatives (Arjunolic acid, Noscapine) were also investigated which are known to have antiviral property but whose effect in SARS-CoV-2 has not been studied yet [17–19].

Organization of the paper:

Section 2 covers the methodology used to screen the medicinal compounds, acquire protein structures from databases, and methodology used to carry out the molecular docking, Sect. 3 covers the results and gives a brief account of the compounds with best scores, Sect. 4 includes a discussion on the result and comparative analysis, and Sect. 5 covers the conclusion.

2 Methodology

2.1 Data Collection and Preparation

1. Protein structure selection and preparation

The crystal structure of RBD of SARS-CoV-2 complexed with human ACE2 was obtained from Protein Data Bank (PDB) with the id 6MOJ. In this study, the interactions were studied with both the proteins separately and in the complex as well. Both the chains in the structure were separated and saved in PDB format after visualization in PyMOL. All the additional ligands present in the structure, that may hinder the interaction between the receptor and the compounds, were removed. The entire preparation protocol of all the protein targets files (Spike protein, ACE2, and complex of spike-ACE2) were done using AutoDock Vina and MGL tools 1.1.2 in order to optimize the protein receptors for docking. Further preparation of the protein receptors was done by deletion of the water molecules, the addition of the polar hydrogens, Kollman charges, Gasteiger charges. Atoms were assigned as AD4 type before the files were finally converted and saved in .PDBQT format [20].

Table 1 List of compounds used in this study

Compounds	Compounds	Compounds	Compounds
9-Bromo-noscapine	Chloroquine	Isoflavone	Nelfinavir
Aloe-emodin	Curcumin	Kaempferol	Noscapine
Amaranthin betacyanin	Demethoxycurcumin	Licoleafol	Oleuropein
Amino-noscapine	Dihydromyricetin	Lopinavir	Quercetin
Apigenin-7-glucoside	Enoxacin	Luteolin-7-glucoside	Remdesivir
Arjunolic acid	Epicatechin gallate	Methyl rosmarinate	Rhein
Calceolarioside B	Eriodictyol	Myricitrin	Th_isoflavone
Catechin	Hydroxychloroquine	Naringenin	Withaferin A
			Withanolide D

2. Ligand Selection and Preparation

The compounds used in this study were selected after an extensive literature survey of phytochemicals compounds and compounds from medicinal plants pertaining to respiratory disorders caused by SARS-CoV and other viruses. The compounds were selected based on their source, their properties, and already known usage to treat respiratory diseases. All the ligands selected for this study are phytochemicals and those which are derived from medicinal plants. In addition, two more compounds (Arjunolic acid and Noscapine) and their derivatives were studied based upon a literature survey whose effect in SARS-CoV-2 has not been studied yet but they have been investigated to have antiviral properties. The ligands used in this study are listed in Table 1.

The 3D structures of the ligands were obtained from Pubchem in SDF format. The 3D SDF structure of all the compounds was converted into the .PDBQT format for the molecular docking using AutoDock Vina after root detection.

2.2 Binding Site Selection and Preparation

The information of the binding residues between the ACE2 and Spike was obtained from the recent studies done on the complex structure. The proposed binding site on the ACE2 receptor comprised of residues Lys31, Glu35, Asp38, and Lys353, binding site on the RBD of spike protein consisted of residues Lys417, Gly446, Tyr449, Tyr453, Leu455, Phe456, Ala475, Phe486, Asn487, Tyr489, Gln493, Ser494, Gly496, Gln498, Thr500, Asn501, Gly502, and Tyr505.¹³ For the complex of both the proteins, after searching the literature, it was found that interface residues between the ACE2 and Spike protein in complex form coincides with the residues targeted on ACE2 and spike protein individually with additional surrounding residues. Therefore, this information was utilized for the interaction with the complex.

2.3 Molecular Docking

After the preparation of the receptor proteins and the ligands, the molecular docking was performed in batches, one for each receptor protein using AutoDock Vina version 1.1.2. For ACE2 receptor, two hotspots (Lys 31 and Lys 353) and the surrounding residues Glu35, Asp38 were targeted and utilized for receptor grid box generation with the dimensions of 40Å X 48Å X 32Å with the spacing of 0.447Å centering around the residues.

On the spike protein, the residues in the core of the receptor-binding domain were targeted. A receptor grid box was set with dimensions 32Å X 60Å X 48Å with a spacing of 0.425Å centering around the residues Lys417, Gly446, Tyr449, Tyr453, Leu455, Phe456, Ala475, Phe486, Asn487, Tyr489, Gln493, Ser494, Gly496, Gln498, Thr500, Asn501, Gly502, and Tyr505.

For the complex of ACE2 in humans and spike protein of SARS-CoV-2 the interface residues important for binding of the two proteins in the complex were targeted. The selected residues are as follows—ACE2 receptor [Phe28, Asp30, Lys31, His34, Glu35, Glu37, Asp38, Leu79, Asn330, Lys353, Arg357, Asp 393] and on Spike protein [Lys417, Gly446, Tyr449, Tyr453, Phe456, Ala 475, Phe486, Gln493, Tyr489, Gly496, Gln498, Thr500, Asn501, Gly502, Tyr505]. A grid box with dimensions 48Å X 66Å X 66Å with a spacing of 0.375Å was set centering around the residues. The docking with the receptor and the ligand was performed using AutoDock Vina. The results were analyzed by using PyMol and LigPlot plus v2.2 manually to study the interactions.

3 Results

After successfully obtaining the PDB structure of the complex of ACE2 and Spike protein of SARS-CoV-2, the protein receptor files to be used for this study were prepared after followed by successfully carrying out the optimization of protein receptors and saving them in PDBQT using AutoDock Vina 1.1.2 and MGL tools. The inhibitors used in this study were successfully downloaded from PubChem and were optimized in AutoDock Vina 1.1.2.

The binding site residues selected after extensive literature reading were loaded in autodock. It was observed that the binding site of both the proteins taken individually from the literature coincides with the residues present in the binding interface of both the proteins in the complex form. After the selection of binding site residues, the grid boxes were generated efficiently for all three protein receptors.

Molecular docking was carried out successfully using AutoDock Vina 1.1.2. The protein receptor and the resulted conformation of the protein were assembled into one file using PyMOL and then were visualized in ligplot plus v 2.2 to evaluate the interaction of the inhibitor with amino acids of the proteins. In analyzing the result,

the binding affinity and the targeted amino acid residues, as well as the hydrogen bonds, were taken into consideration.

3.1 Molecular Docking of Spike's Receptor Binding Domain with the Compound Dataset

In the present study, after study the interactions of spike receptor-binding domain of SARS-CoV-2 with 33 number of compounds, binding affinity ranged from -8.1 to -5.1 kcal/mol as given in the table below (Table 2). The results were analyzed in PyMol and LigPlot plus, it was found that majorly eight compounds bound well to the receptor-binding domain of spike protein in SARS-CoV-2. The controls taken in this study, Lopinavir and Nelfinavir showed lower binding energy of -6.6 kcal/mol and -6.3 kcal/mol, respectively, in comparison to these eight compounds. This depicts that these compounds performed better than the controls and might have greater potential to become inhibitors than the existing inhibitors.

The best interacting compounds displaying the highest binding affinity were Withanolide D (-8.1 kcal/mol), Amaranthin betacyanin (-7.5 kcal/mol). The Arjunolic

Table 2 List of compounds and their binding energy (Kcal/mol) in docking with Spike protein in SARS-CoV-2

Compounds	B.E (Kcal/mol)	Compounds	B.E (Kcal/mol)
Withanolide D	-8.1	9-bromo-noscapine	-6.5
Amaranthin betacyanin	-7.5	Isoflavone	-6.5
Luteolin-7-glucoside	-7.4	Th_isoflavone	-6.5
Apigenin-7-glucoside	-7.3	Naringenin	-6.4
Calceolarioside B	-7.2	Curcumin	-6.3
Withaferin A	-7	Enoxacin	-6.3
Remdesivir	-7	Kaempferol	-6.3
Licoleafol	-6.9	Nelfinavir	-6.3
Myricitrin	-6.9	Aloe-emodin	-6.2
Dihydromyricetin	-6.8	Amino-noscapine	-6.2
Epicatechingallate	-6.8	Noscapine	-6.2
Eriodictyol	-6.8	Oleuropein	-6.2
Quercetin	-6.8	Demethoxycurcumin	-6
Catechin	-6.7	Methyl rosmarinate	-5.7
Arjunolic acid	-6.7	hydroxychloroquine	-5.2
Lopinavir	-6.6	Chloroquine	-5.1
Rhein	-6.6		

acid displayed binding energy of -6.7 kcal/mol and Noscapine displayed binding energy of -6.2 kcal/mol.

Withanolide D displayed the highest binding energy of -8.1 kcal/mol. It forms seven hydrogen bonds with the residues Arg403, Tyr453, Gly496, Gln498 and has hydrophobic interactions with Lys417, Tyr495, Tyr505, and Asn501 (Fig. 1a). Amaranthin betacyanin displayed the binding energy of -7.5 kcal/mol. It forms nine hydrogen-bonded interactions with residues Arg403, Tyr453, Glu484, Gln493, Ser494, Gly496, Gln498, Tyr505 on the Spike protein and has hydrophobic bonded interactions with Tyr449, Phe490, Tyr495, Phe497, and Asn501 (Fig. 1b). Arjunolic acid displayed a binding energy of -6.7 kcal/mol. It forms only one hydrogen bond with Ser494 and multiple hydrophobic bonds with Tyr495, Gly496, Gln498, Tyr505,

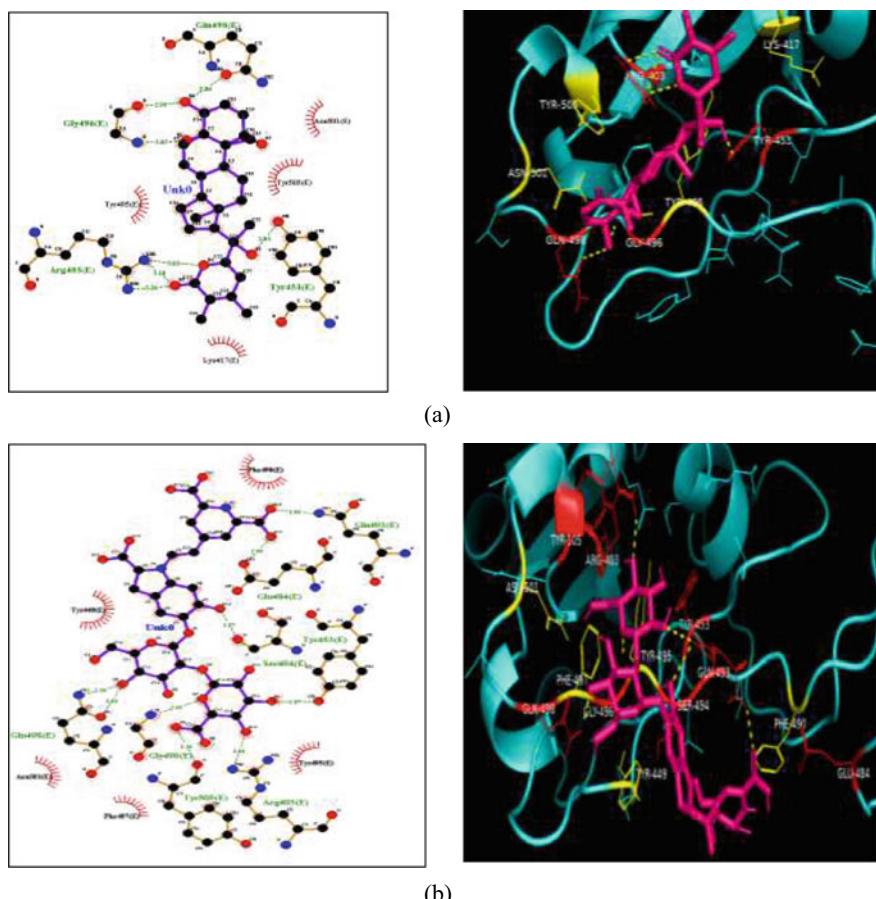


Fig. 1 Interactions of the compounds with the the highest binding energies with the Spike protein of SARS-CoV-2. The red-colored residues indicate the hydrogen-bonded interactions and yellow-colored residues indicate the hydrophobic interactions. **a** Withanolide D **b** Amaranthin betacyanin

Asn501, and Gly502. With binding energy of -6.2 kcal/mol, Noscapine formed one hydrogen bonded interaction with Tyr449 and numerous hydrophobic bonded interactions with Tyr453, Tyr495, Gly496, Phe497, Gln498, Asn501, Tyr505.

3.2 Molecular Docking of ACE2 in Humans with the Compound Dataset

After studying the interactions of ACE2 with 33 number of compounds, using molecular docking, it was found that the binding affinities were in the range of -9.9 to -6.8 kcal/mol, as given in the table (Table 3), which were comparable with the already used compounds Lopinavir and Nelfinavir.

The compounds with the highest binding affinity among them are Withanolide D (-9.9 kcal/mol), Arjunolic acid (-9.1 kcal/mol), Calceolarioside B (-9 kcal/mol). The binding affinity of Noscapine was -7.4 kcal/mol. Within the binding site, these compounds formed multiple hydrogen bonds and hydrophobic bonds with the Ser44, Ser47, Lys74, Ala348, Leu391, Asn394, Arg393, His401. The controls Lopinavir and Nelfinavir had the binding energy -9.4 kcal/mol and -9 kcal/mol, respectively. Lopinavir formed two hydrogen bonds with the residue Arg394 and

Table 3 List of compounds and their binding energy (Kcal/mol) in docking with ACE2 found in humans

Compounds	B.E (Kcal/mol)	Compounds	B.E (Kcal/mol)
Withanolide D	-9.9	Amino-noscapine	-7.7
Lopinavir	-9.4	Dihydromyricetin	-7.7
Arjunolic acid	-9.1	Curcumin,	-7.7
Calceolarioside B	-9	Th_isoflavone	-7.7
Withaferin A	-9	Eriodictyol	-7.6
Nelfinavir	-9	Aloe-emodin	-7.5
Amaranthin betacyanin	-8.8	Catechin	-7.5
Epicatechingallate	-8.7	9-bromo-noscapine	-7.4
Luteolin-7-glucoside	-8.6	Noscapine	-7.4
Apigenin-7-glucoside	-8.4	Kaempferol	-7.3
Licoleafol	-8.1	Methyl rosmarinate	-7.3
Remdesivir	-8	Naringenin	
Rhein	-8	Isoflavone	-7.1
Oleuropein	-8	Enoxacin	-6.8
Demethoxycurcumin	-7.8	Hydroxychloroquine	-6.2
Myricitrin	-7.8	Chloroquine	-6
Quercetin	-7.8		

interacted with other residues such as Phe40, Ser44, Trp69, Leu73, Ala99, Thr347, Trp349, Asp350, His378, Ala382, Phe390, Lue391, Arg393. His401 via hydrophobic interaction. Nelfinavir did not form hydrogen bond with any residue. It interacted via hydrophobic bonded interactions with Phe40, Ser44, Ala348, Trp349, Asp350, His378, Phe390, Arg393, His401.

Withanolide D gave the top score with the binding energy of -9.9 kcal/mol, forming hydrogen interactions with Asn394 and Lys74 with hydrophobic bonded interactions with Phe40, Ser70, Ser77, Leu73, Ala99, Arg393, Phe39 (Fig. 2a). Arjunolic acid with the binding energy of -9.1 kcal/mol makes key hydrophobic interactions with Ser77, Phe40, Trp69, Arg393 and forms one hydrogen bond with Leu391 amino acid (Fig. 2b). Calceolarioside B has the binding energy -9 kcal/mol, forms ten Hydrogen bonds with Leu391, Tyr385, His401, Asp382, Ala348, Ser44, Ser47 and has hydrophobic interactions with Phe40, Trp349, Arg393 Asp350, Phe390 depicts affinity for hotspot 353 (Fig. 2c)

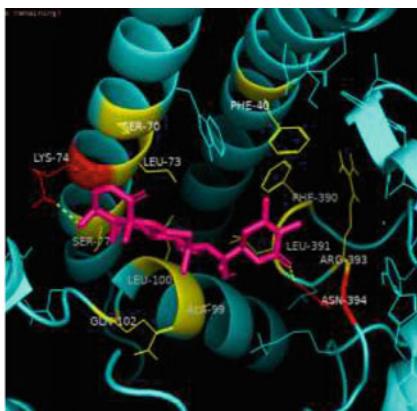
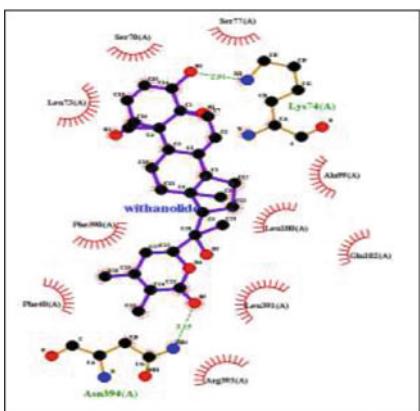
Noscapine with the binding energy of -6.8 kcal/mol. It forms two hydrogen bonds with Asp350, Ser4 and forms multiple hydrophobic interactions with Phe40, Ser44, Tyr347, Ala348, Trp349, Asp382, Tyr385, and His401.

3.3 Molecular Docking of the Complex of the Spike Protein of SARS-CoV-2 and ACE2 with the Compound Dataset

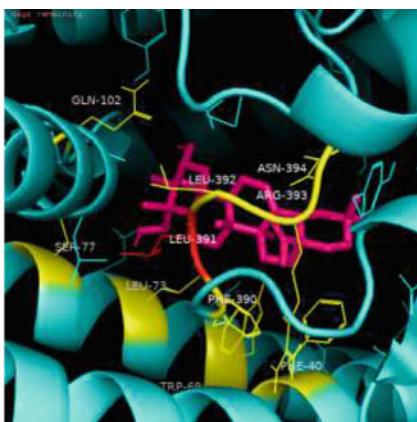
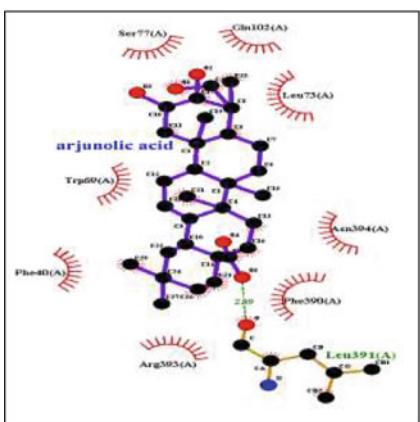
After studying the interactions of the complex with 33 compounds, it was found that the binding energy ranged from -9.9 to -6.2 kcal/mol, as given in the table below (Table 4). After analyzing the results using PyMOL and LigPlot plus, eight compounds were identified which have a high affinity as well as bind to the targeted residues in the complex.

The best of the list compounds with the minimum binding energy were Withanolide (-9.9 kcal/mol), Amaranthin betacyanin (-9.3 kcal/mol), Arjunolic acid (-9.2 kcal/mol), Calceolarioside (-9.3 kcal/mol), Withaferin (-8.9 kcal/mol) and Oleuropein (-8.4 kcal/mol). The controls taken for this study were Lopinavir (-9.0 kcal/mol) and Nelfinavir (-8.8 kcal/mol). Lopinavir forms 2 hydrogen bonds with Asn394 residue and forms many hydrophobic bonds with Phe40, Trp69, Leu73, Ala99, Asp350, Phe390, Leu391. Nelfinavir only forms hydrophobic binds with Phe40, Ser44, Ala348, Trp349, Asp350, His378, Phe390, Arg393, Asn394, His401.

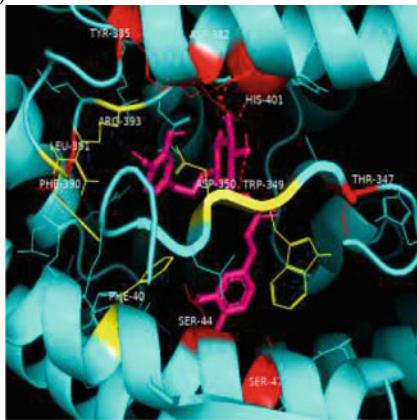
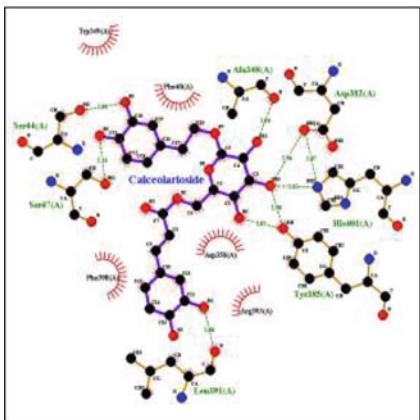
The top four compounds are (i.e. Withanolide D (-9.9 kcal/mol), Amaranthin betacyanin (-9.3 kcal/mol), Arjunolic acid (-9.2 kcal/mol), Withaferin A (-8.9 kcal/mol) with highest binding energy. These compounds form multiple bonds as well as have interactions with the residues that support the inhibition of the contact of the virus and ACE2. Withanolide forms two hydrogen bonds with Lys74 and Asn394 and non-bonded interactions with multiple residues such as Phe40, Ser70, Leu73, Ser77, Ala99, Leu100, Leu391, Phe390, Arg393 (Fig. 3a). Amaranthin, though with a binding energy of -9.3 kcal/mol, didn't show interaction with any



(a)



(b)



(c)

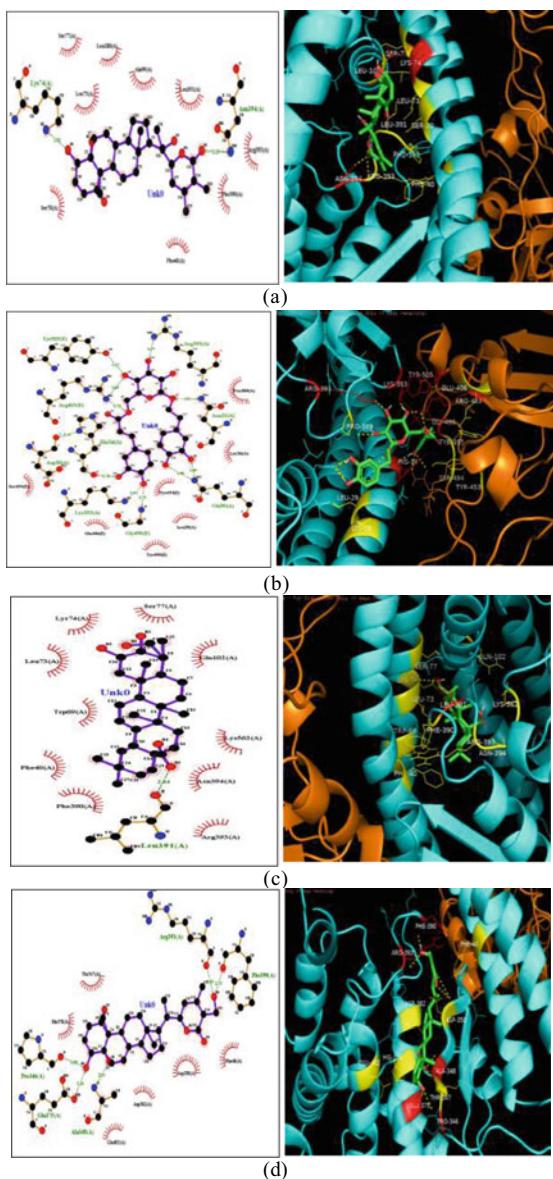
Fig. 2 Interactions of the compounds with the highest binding energies with ACE2. The red-colored residues indicate the hydrogen-bonded interactions and yellow-colored residues indicate the hydrophobic interactions. **a** Withanolide D, **b** Arjunolic acid, **c** Calceolarioside

Table 4 List of compounds and their binding energy (Kcal/mol) in docking with complex

Compounds	B.E (Kcal/mol)	Compounds	B.E (Kcal/mol)
Withanolide D	-9.9	Kaempferol	-8
Amaranthin betacyanin	-9.3	Dihydromyricetin	-7.9
Calceolarioside B	-9.3	Myricitrin	-7.9
Arjunolic acid	-9.1	Curcumin	-7.8
Lopinavir	-9	Naringenin	-7.8
Withaferin A	-8.9	Remdesivir	-7.8
Nelfinavir	-8.8	Aloe-emodin	-7.7
Apigenin-7-glucoside	-8.7	Noscapine	-7.7
Rhein	-8.7	Amino-noscapine	-7.7
Luteolin-7-glucoside	-8.6	9-bromo-noscapine	-7.6
Oleuropein	-8.4	Enoxacin	-7.5
Th_isoflavone	-8.4	Methyl rosmarinate	-7.4
Eriodictyol	-8.3	Isoflavone	-7.2
Quercetin	-8.3	Demethoxycurcumin	-7.1
Epicatechin gallate	-8.2	Hydroxychloroquine	-6.3
Catechin	-8.1	Chloroquine	-6.2
Licoleafol	-8.1		

significant residue in the interface of the complex. Amaranthin forms nine hydrogen bonds with Gln81, Leu85, Gln86, Gln101, Tyr196, Tyr202, Glu208, Asn210, and Lys562 and form hydrophobic interactions Tyr83, Pro84, Leu95, Gln98, Gln102, Asn103 and Val209 (Fig. 3a). Calceolarioside with binding energy -9.3 kcal/mol, formed the nine hydrogen bonds with Tyr505, Arg403, Gly496, Lys353, Arg393, His34, Gln96 with hydrophobic interactions with Pro389, Lys36, Tyr453, Leu29, Glu406, and Ser494. Calceolarioside inhibits the binding of spike and ACE2 by binding with many target residues on both the spike protein and ACE2 receptor (i.e. Tyr505, Gly496, Ser494, Tyr453 on the spike and His34, Asp30, Lys353, Arg393 on ACE2 receptor) (Fig. 3b). Arjunolic acid makes one hydrogen bond with Leu391 and makes several non-bonded interactions with residues such as Phe40, Trp69, Leu73, Lys74, Ser77, Gln102, Phe390, Lys562, Arg393, Asn394 (Fig. 3c). Withaferin forms five hydrogen bonds with the residues (Pro346, Glu375, Ala348, Phe390, Arg393) and has many favorable hydrophobic interactions (Fig. 3d). Another compound, Oleuropein (-8.4 kcal/mol) interact and form a hydrogen bond with Lys353, Asp30, Gly496, Arg403 that are the key residues of interface important for the interaction between the Spike protein of SARS-CoV-2 and ACE2. It also forms several hydrophobic non bonded interactions with residues Ser494, Glu37, Tyr453, Tyr495, Asp38, Glu406, Asn33, Lys417, Leu29.

Fig. 3 Interactions of the compounds with the highest binding energies with the complex of spike protein and ACE2. The chains in orange represent the spike protein of SARS-CoV-2 and the chains in cyan blue represent the ACE2. The red-colored residues indicate the hydrogen-bonded interactions and yellow-colored residues indicate the hydrophobic interactions. **a** Withanolide D, **b** Calceolarioside B, **c** Arjunolic acid, **d** Withaferin



4 Discussion

The study is focused on finding the potential compounds which can inhibit the contact between the SARS-CoV-2 and ACE2 in the host. In this study, a set of medicinal compounds exhibiting antiviral properties were computationally analyzed

for interaction with the RBD domain of spike protein of SARS-CoV-2 and ACE2 receptor of the host. We have used in silico approach to find out the interactions of the compound dataset with the RBD domain of SARS-CoV-2 and ACE2 receptor both in complex form and individually. The aim was to identify the compounds which show high binding affinity with either of the receptors or the complex in order to further explore the useful information to prevent the interaction of the RBD domain of SARS-CoV-2 and ACE2 receptor of the host. We have identified certain compounds that are potentially promising in this direction. The compounds with the highest binding energy that show high affinity for the RBD domain of SARS-CoV-2 are Withanolide D (-8.1 kcal/mol), Amaranthin betacyanin (-7.5 kcal/mol), Apigenin-7-glucoside (-7.4 kcal/mol), Calceolarioside B (-7.2 kcal/mol), Withaferin A (-7 kcal/mol), Remdesivir (-7 kcal/mol) and licoleafol (-6.9 kcal/mol). All these compounds form several hydrogen bonds with the contact residues on the spike receptor-binding domain and have many hydrophobic interactions with the nearby residues that support the interaction between the ligand and the spike receptor-binding domain protein of SARS-CoV-2. These compounds display the potential to disrupt the contact of the virus SARS-CoV-2 and the host receptor ACE2 by binding with the RBD of the spike protein. The controls taken in this study were Nelfinavir with binding energy -6.3 kcal/mol and Lopinavir with binding energy -6.6 kcal/mol. This indicates that the compounds identified performed better than the controls.

The compounds with the highest binding energy that show affinity for the ACE2 are Withanolide D (-9.9 kcal/mol), Arjunolic acid (-9.1 kcal/mol), Calceolarioside B (-9 kcal/mol), Amaranthin betacyanin (-8.8 kcal/mol), Epicatechin gallate (-8.7 kcal/mol), Luteolin-7-glucoside (-8.6 kcal/mol), Apigenin-7-glucoside (-8.4 kcal/mol), Licoleafol (-8.1 kcal/mol), Dihydromyricetin (-7.7 kcal/mol). The controls taken were Lopinavir and Nelfinavir with binding energy as -9.4 and -9 kcal/mol. This depicts that Withanolide D performs better than both the controls and Arjunolic acid, Calceolarioside B performed better than Nelfinavir. And other compounds show the affinity for the ACE2 and interacts with the receptor via hydrophobic as well as hydrogen bonds. Except for the Arjunolic acid and Withanolide D, the other compounds form more than two hydrogen bonds with the ACE2 whereas the control Nelfinavir forms none, and Lopinavir forms only one. Forming several hydrogen bonds and the hydrophobic interactions with the key residues on the ACE2 hypothesized to form contact with spike, these compounds show potential to disrupt the contact with the spike protein on the SARS-CoV-2, by binding to the ACE2 receptor.

The compounds Withanolide (-9.9 kcal/mol), Amaranthin betacyanin (-9.3 kcal/mol), Arjunolic acid (-9.2 kcal/mol), Calceolarioside B (-9.3 kcal/mol), Withaferin (-8.9 kcal/mol), and Oleuropein (-8.4 kcal/mol) showed the highest affinity for the complex of ACE2 and spike protein of SARS-CoV-2. Out of these, Calceolarioside B and oleuropein interacted with the hotspot residue Lys353 on the ACE2- spike protein interface thus exhibiting the inhibition of the interaction of the spike protein and ACE2.

In this study, the interactions of the currently used drugs, Chloroquine, Hydroxychloroquine, Remdesivir were also investigated with all three protein receptors

and these compounds have a lower binding affinity in comparison with identified compounds of this study. This established the potential of the compounds identified in this study for further research in this direction.

5 Conclusion

In conclusion, the compounds Withanolide D, Amaranthin betacyanin, Calceolarioside B, Withaferin performed well in the docking studies. Out of Arjunolic acid, Noscapine and its derivatives, Arjunolic acid displayed high affinity for all the receptors than Noscapine and its derivatives. It seems that having a higher affinity to the spike protein, ACE2 and the complex of the two, these compounds have the potential and capability to become good agents that can inhibit the interaction of the spike of SARS-CoV-2 and ACE2 in the host. Hence, these compounds are recommended compounds as the inhibitors of SARS-CoV-2 and its interaction with ACE2 and should be further explored in future research.

References

1. Archived: WHO Timeline-COVID-19 <https://www.who.int/news-room/detail/27-04-2020-who-timeline-covid-19>. Accessed 24 Oct 2020
2. WHO Coronavirus Disease (COVID-19) Dashboard|WHO Coronavirus Disease (COVID-19) Dashboard https://covid19.who.int/?gclid=CjwKCAjwoc_8BRAcEiwAzJevtRLrSM0E-GDJuHxdtOFb53uDuVDrRh0fSTiUMrt0rXVi9anwEi5oHRoC9NkQAvD_BwE. Accessed 14 Dec 2020
3. Y. Chen, Q. Liu, D. Guo, Emerging Coronaviruses: genome structure, replication, and pathogenesis. *J. Med. Virol.* John Wiley and Sons Inc., 418–423. <https://doi.org/10.1002/jmv.25681>. Accessed 1 Apr 2020
4. E. Sakalli, D. Temirkov, E. Bayri, E.E. Alis, S.C. Erdurak, M. Bayraktaroglu, Ear nose throat-related symptoms with a focus on loss of smell and/or taste in COVID-19 patients. *Am. J. Otolaryngol. Head Neck Med. Surg.* **41**(6), 102622 (2020). <https://doi.org/10.1016/j.amjoto.2020.102622>
5. Y. Yuan, D. Cao, Y. Zhang, J. Ma, J. Qi, Q. Wang, G. Lu, Y. Wu, J. Yan, Y. Shi, X. Zhang, G.F. Gao, Cryo-EM structures of MERS-CoV and SARS-CoV spike glycoproteins reveal the dynamic receptor binding domains. *Nat. Commun.* **8** (2017). <https://doi.org/10.1038/ncomms15092>
6. W. Ni, X. Yang, D. Yang, J. Bao, R. Li, Y. Xiao, C. Hou, H. Wang, J. Liu, D. Yang, Y. Xu, Z. Cao, Z. Gao, Role of Angiotensin-Converting Enzyme 2 (ACE2) in COVID-19. *Critical Care* (BioMed Central), 422. <https://doi.org/10.1186/s13054-020-03120-0>. Accessed 13 July 2020
7. H. Zhang, J.M. Penninger, Y. Li, N. Zhong, A.S. Slutsky, Angiotensin-Converting Enzyme 2 (ACE2) as a SARS-CoV-2 receptor: molecular mechanisms and potential therapeutic target. *Intensive Care Med.* **46**(4), 586–590 (2020). <https://doi.org/10.1007/s00134-020-05985-9>
8. J. Liu, X. Liao, S. Qian, J. Yuan, F. Wang, Y. Liu, Z. Wang, F.S. Wang, L. Liu, Z. Zhang, Community transmission of severe acute respiratory syndrome coronavirus 2, Shenzhen, China, 2020. *Emerg. Infect. Dis.* **26**(6), 1320–1323 (2020). <https://doi.org/10.3201/eid2606.200239>

9. Y. Huang, C. Yang, X. Xu, W. Xu, S.W. Liu, Structural and functional properties of SARS-CoV-2 spike protein: potential antivirus drug development for COVID-19. *Acta Pharmacologica Sinica* (Springer Nature), 1141–1149 (2020). <https://doi.org/10.1038/s41401-020-0485-4>. Accessed 1 Sept 2020
10. G.K. Veeramachaneni, V.B.S.C. Thunuguntla, J. Bobbillapati, J.S. Bondili, Structural and simulation analysis of hotspot residues interactions of SARS-CoV 2 with human ACE2 receptor. *J. Biomol. Struct. Dyn.* 1–11 (2020). <https://doi.org/10.1080/07391102.2020.1773318>
11. K. Wu, W. Li, G. Peng, F. Li, Crystal structure of NL63 respiratory coronavirus receptor-binding domain complexed with its human receptor. *Proc. Natl. Acad. Sci. U.S.A.* **106**(47), 19970–19974 (2009). <https://doi.org/10.1073/pnas.0908837106>
12. Y. Wan, J. Shang, R. Graham, R.S. Baric, F. Li, Receptor recognition by the novel coronavirus from Wuhan: an analysis based on decade-long structural studies of SARS Coronavirus. *J. Virol.* **94**(7), 127–147 (2020). <https://doi.org/10.1128/jvi.00127-20>
13. J. Lan, J. Ge, J. Yu, S. Shan, H. Zhou, S. Fan, Q. Zhang, X. Shi, Q. Wang, L. Zhang, X. Wang, Structure of the SARS-CoV-2 spike receptor-binding domain bound to the ACE2 receptor. *Nature* **581**(7807), 215–220 (2020). <https://doi.org/10.1038/s41586-020-2180-5>
14. R. Subramani, M. Narayanasamy, K.D. Feussner, Plant-derived antimicrobials to fight against multi-drug-resistant human pathogens. *Biotech. Springer Verlag*, 1–15 (2017). <https://doi.org/10.1007/s13205-017-0848-9>. Accessed 1 July 2017
15. K. Mohanraj, B.S. Karthikeyan, R.P. Vivek-Ananth, R.P.B. Chand, S.R. Aparna, P. Mangalapandi, A. Samal. IMPPAT: a curated database of indian medicinal plants, phytochemistry and therapeutics. *Sci. Rep.* **8**(1) (2018). <https://doi.org/10.1038/s41598-018-22631-z>
16. S. Adem, V. Eyupoglu, I. Sarfraz, A. Rasul, M. Ali, Identification of potent COVID-19 main protease (Mpro) inhibitors from natural polyphenols: an in silico strategy unveils a hope against CORONA. *Preprints*. <https://doi.org/10.20944/preprints202003.0333.v1>. Accessed 23 Mar 2020
17. M.T. Ul Qamar, S.M. Alqahtani, M.A. Alamri, L.L. Chen, Structural basis of SARS-CoV-2 3CLpro and Anti-COVID-19 drug discovery from medicinal plants. *J. Pharm. Anal.* **10** (4), 313–319 (2020). <https://doi.org/10.1016/j.jpha.2020.03.009>
18. NOPR: Arjunolic acid: A novel phytomedicine with multifunctional therapeutic applications. <http://nopr.niscair.res.in/handle/123456789/7399>. Accessed 18 Oct 2020
19. S.A. Ebrahimi, Noscapine, a Possible drug candidate for attenuation of cytokine release associated with SARS-CoV-2. *Drug Develop. Res.* Wiley-Liss Inc. (2020). <https://doi.org/10.1002/ddr.21676>
20. O. Trott, A.J. Olson, AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.* **31**(2) (2009). <https://doi.org/10.1002/jcc.21334>

Serverless Computation with NuLambda



R. Eashwaran, Tanya Mittal, and Kavita Sheoran

Abstract In recent years, the use of serverless computing and, in particular, Function as a Service (FaaS) has increased as an execution model where there is no clear server management performed by the user as on virtual machines. Instead, the cloud provider dynamically allocates resources to the function invocations and a corresponding charge is introduced depending on the execution time and shared memory, as demonstrated by AWS Lambda. We are launching nuLambda, which is an open-source platform for building future apps. We provide some features of serverless computation also the many research challenges that need to be addressed in the use of these systems for a microservice event-driven architecture design with services that meet freely and transparently. We described a deployment architecture such that individual components scale elastically and developed a thread-safe Docker-based sandboxing tool to safely execute user code. We also included a brief study of web applications that has been added to better motivate other features of serverless application construction.

Keywords Lambda · Serverless computation · Apache kafka · Function as a service

R. Eashwaran (✉) · T. Mittal · K. Sheoran
Maharaja Surajmal Institute Of Technology, New Delhi, India
e-mail: kavita.sheoran@msit.in

1 Introduction

The fast movement in the advancement of data centers and the software platforms is set to bring a change in how we handle our online applications and services. Earlier, applications used to run on a physical machine which solely belonged to it. The significant expenses of purchasing and keeping up huge quantities of machines, and the way that each was frequently underutilized, prompted virtualization which extraordinarily lessened costs and improved reasonability. Notwithstanding, equipment-based virtualization isn't a solution for all challenges, and lighter-weight innovations have emerged to address its key issues.

One way to do this is using containers. And using Docker, the containers let engineers to quickly deploy and test new services without the slow delivery and operating time of virtual machines [1]. The purpose of using Docker was the level of support and usability that provided the performance of microservices. We may contain services and manage each measurement. Besides, Docker helped ensure that features and improvements would be seamlessly transmitted between development and production teams. Within these processes, services have moved smoothly with high availability and durability.

However, as the number of services and the complexity of each service increased, it became clear that we needed a way to increase computing power. The traffic we get through these services is completely unexpected. But if possible, the service should be available and able to handle the increasing burden especially of third-party integration where the data is synced. During this time, there were success stories that reflect what we have been looking for as a SaaS solution provider. Being able to remove all concerns was a great relief, especially when combined with the added benefit of paying for only what the person uses. So, we started moving our services to a server without a server. Serverless Computation has a great way of providing and updating the resources needed to run non-server apps. Instead of deploying of applications as server, build them as a set of functions with a common datastore.

1.1 *Lambda*

Lambda is a computer service without servers. Lambda users perform tasks, self-contained applications in a supported language and operating time, then put them on Lambda, which performs those functionalities efficiently and flexibly. Lambda functions can execute any type of computer function, from running web pages and continuous data streams to calling APIs and integration with other utilities. The concept of a serverless computer refers to the need to keep your servers performing these tasks. Lambda is a completely managed service that manages all your equipment. So, serverless doesn't imply there are no servers included, it just means that servers, operating systems, network layers, and all the infrastructure are already taken care of so one can focus on coding [2].

Each Lambda function works in its container. When the work is done, Lambda puts it in a new container and removes that container from a collection of equipment that employs many people outside. Before operations begin, each task container is assigned a RAM and CPU capacity. When tasks are completed, initially allocated RAM is repeated while the task you used is running. Customers are then charged based on shared memory and duration of work taken to complete. Users don't get much visibility of how the system works, but they also don't have to be concerned about updating sub-devices, avoiding network conflicts, and so on. Lambda takes care of itself. Since the service is completely managed, utilizing Lambda leads to saving one's time on performance tasks. In case there is no infrastructure to take care of, you can spend a lot of time executing with the program code even if this means you stop the flexibility of using your infrastructure.

When one uses Lambda, they are only responsible for their code. Lambda controls a compute fleet that provides a balance of memory, CPU, network, and other resources. These issues allow Lambda to perform operational and administrative functions on the user's behalf, including the ability to provide, monitor fleet health, security, deploying code, and keeping a track of their Lambda activities. One of the unique features of Lambda is that many events of similar or not same functions from the same account can be performed simultaneously. Besides, the fee may vary depending on the time, and such variations cause no effect to Lambda; a person has to only pay for the use of computer services. This makes Lambda good to use for deploying cloud computing solutions.

2 Literature Survey

A literature review revealed research about some designs and implementations of components of the FaaS platform.

According to Wagner [3], AWS Lambda functions perform in a container that separates them from other functions and provides resources, such as memory, defined in the configuration. This document discusses how Lambda builds and uses these sandboxes, as well as the impact of those policies on the system model.

Dynamic languages like JavaScript are much harder to integrate than statically typed. Since no concrete type information is available, traditional compilers need to generate a standard code that can handle all combinations of the type available during operation. Andreas et al. [4] introduced another combination of typed languages that dynamically identifies a loop track that is routinely used during operation and then generates machine code in time for the actual dynamic types that occur in each loop line. Our approach offers a unique approach to the type of process to be followed, as well as a more efficient and effective way to further integrate alternatives that have been acquired lazily through the included loops. We have used a powerful JavaScript compiler based on our expertise and measured 10x speeds and more on specific scaling systems.

Harter et al. [5] describes Slacker, a new Docker storage driver designed for the start of a container. Slacker is based on shared storage between all Docker staff and registers. Employees provide faster storage of containers using backend clones and reduce start-up delays by downloading container data. Slacker reduces container delivery cycle by 5 and development cycle by 20 times.

Larger data analysis frameworks tend to be shorter in terms of tasks and higher levels of similarity to provide lower latency. Scheduling highly compatible tasks that complete hundreds of milliseconds pose a major challenge to task planners, which will require organizing millions of jobs per second on the appropriate equipment while providing millisecond-level latency and high availability. [6] shows that the fixed, random allocation method provides excellent performance while avoiding the restrictions on entry and availability of a medium-sized design.

3 Proposed Method

We now discuss the components of nuLambda and research problems in the serverless computing space (Fig. 1).

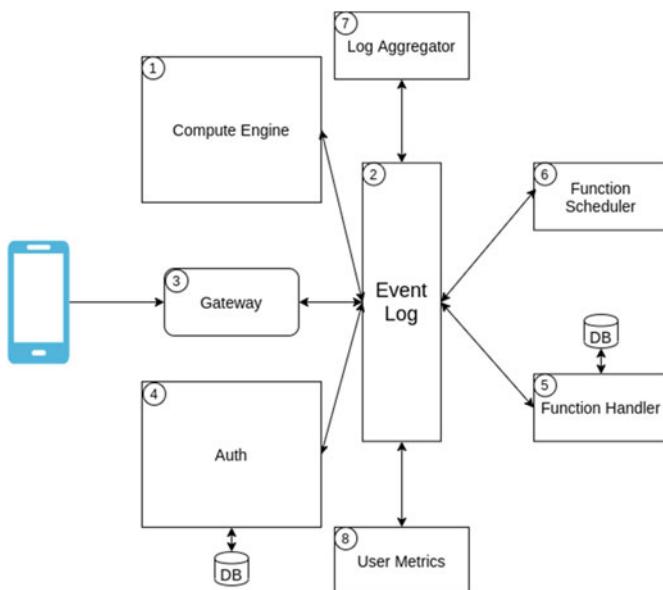


Fig. 1 Microservice architecture of nuLambda

3.1 Compute Engine

A sandbox is at the central part of the construction of the Lambda. AWS Lambda uses sandbox function control containers [3], but avoids any other container-based services by sharing servers and start times among various hosts. On the contrary, nuLambda uses docker containers to sandbox function handlers. The compute engine service listens to any function execute events from the log. For each request, it executes it in a running docker container. If the docker container is paused, then it resumes it before execution. To reduce the cost of delivering containers, AWS Lambda reuses container to make as many maximum number of handlers as possible. Unfortunately, or in this way, Lambdas grow slower than containers at lower input volumes. In nuLambda, a docker container is paused and resumed during low load scenarios. It performs best under high load as the same container is reused again and again and the container resume time is eliminated. Now a container's time to live can be set high to increase reusability but this will lead to high memory usage. When the load is very less, the ping with AWS Lambda is 10x worse than Elastic Beanstalk. The container must be in a working condition to handle requests. In the absence of requests, the container is suspended or stopped. There are delays involved in replacing or reusing a container. Both restart and start-up take 100s milliseconds. In contrast, the break happens in around 1ms. Unfortunately, when containers are suspended, they take up memory. And storing multiple containers suspended will prevent us from building multiple containers. Memory is the largest blocker and temporarily suspended containers place them on top of active containers (Fig. 2).

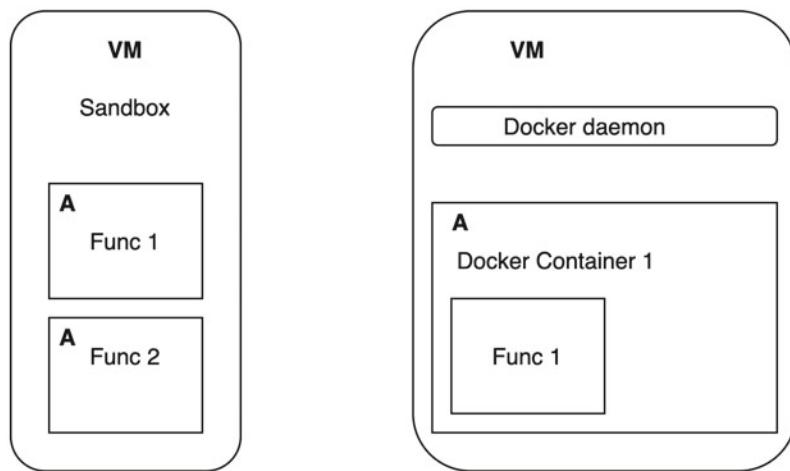


Fig. 2 VM and function instance organization in AWS Lambda (left) and nuLambda (right)

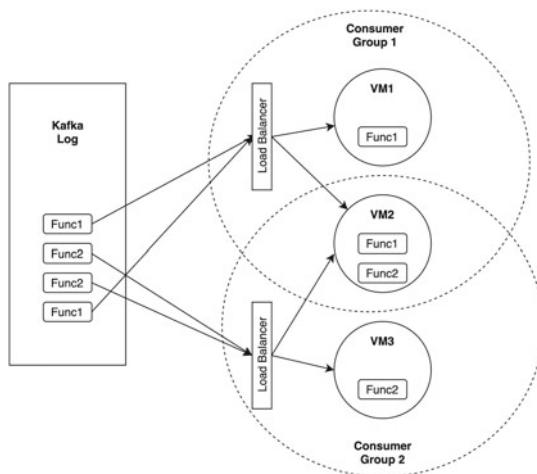
3.2 Language Support

Lambdas are faster because the functions are cached in binaries and these binaries are used frequently. Apparently, advantages fade away when users use many libraries from a third party, as libraries need to be copied over the network at the request of a new Lambda employee. Such wraps can increase start-up delays with the order of magnitude [5]. Lazily cloning dependencies may exacerbate this problem [7]. In nuLambda, we support only JavaScript as the language and only npm as the package repository. This enables us to provide special support and optimizations for executing Javascript functions. The most frequently used packages are stored in each compute engine and the required packages for a function are simply mounted to the node_modules folder inside the container. But there must be a threshold for the package size as large packages will take more disk space. So this kind of package awareness entails new code locality challenges (Fig. 3).

3.3 Load Balancing

Apache Kafka is used as the distributed event log and load balancer in nuLambda. Services register themselves as publishers and/or subscribers to certain topics of concern. Sticky load balancing is required to direct function calls to the same containers to increase reusability. This is achieved by adding all the compute engine instances with suitable containers to a consumer group. The load balancing is done among these consumers in a consumer group. The previous low latency cluster schedulers identify tasks in the range of 100 ms [8]. Lambda schedulers need to schedule a short-order function while considering several local types. First, program planners

Fig. 3 Sticky Load balancing in nuLambda



should consider the location of the session: if the Lambda application is part of the session with an open TCP/IP connection, it will be useful to use a machine manager where the TCP/IP connection is held so that load does not require a proxy transfer. Second, getting the code becomes very difficult [9]. An editor who knows that different handlers depend on common packages can make better deployment decisions. In addition, the developer may want to direct applications based on different levels of dynamic performance acquired by different employees. Third, location data will be important in using Lambdas next to data or large data sets and indicators. The editor will have to wait for what questions will be given to Lambda's specific request, or will read any data. Many new details (e.g., Cassandra [10] and MongoDB [11]) retain symbols like LSM trees.

3.4 Cost Estimation

Previous platforms cannot provide real app costs for individual components/services. For example, applications which are deployed on a virtual machine are usually charged hourly, and it is not clear how you can split the cost in individual applications for more than an hour. Conversely, you may have told exactly how much each RPC person is driving to the Lambda manager calling the cloud customer. The metrics on the basis of which the final cost is calculated are mainly the memory cap, CPU runtime, and disk space cap. Another key metric that could be considered in cost is the user could choose the time-to-live (TTL) for containers, and higher TTL would make the subsequent function calls faster for the given TTL.

4 Conclusion and Future Scope

We provided insights into architectures of modern serverless computing platforms. We discovered several issues, raised from either specific design decisions or engineering, concerning security, performance, and resource accounting in the platforms. We have discussed how using Lambda is more flexible, including container services which autoscale, how this new paradigm brings exciting challenges to execution engines, load balancers, etc. To facilitate research in these areas, we built nuLambda, the implementation of an open-source platform for Lambda. We further plan to build monitoring and logging services that will integrate into existing nuLambda architecture.

References

1. Serverless Computation with OpenLambda, https://www.usenix.org/system/files/conference/hotcloud16/hotelcloud16_hendrickson.pdf
2. AWS Lambda Developer Guide: What is AWS Lambda? <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>
3. T. Wagner, Understanding Container Reuse in AWS Lambda, <https://aws.amazon.com/blogs/compute/container-reuse-in-lambda>
4. A. Gal, B. Eich, M. Shaver, D. Anderson, D. Mandelin, M.R. Haghightat, B. Kaplan, G. Hoare, B. Zbarsky, J. Orendorff, et al., Trace-based just-in-time type specialization for dynamic languages. ACM Sigplan Not. **44**(6), 465–478 (2009)
5. T. Harter, B. Salmon, R. Liu, A.C. ArpaciDusseau, R.H. Arpaci-Dusseau, Slacker: fast distribution with lazy docker containers, in *14th USENIX Conference on File and Storage Technologies (FAST 16)*, (USENIX Association, Santa Clara, CA, 2016), pp. 181–195
6. AWS Developer Forums: Java Lambda Inappropriate for Quick Calls? <https://forums.aws.amazon.com/thread.jspa?messageID=679050>
7. Charity Majors MongoDB RocksDB at Parse, <http://blog.parse.com/announcements/mongodb-rocksdb-parse>
8. K. Ousterhout, P. Wendell, M. Zaharia, I. Stoica, Sparrow: distributed, low latency scheduling, in Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles (ACM, New York, 2013), pp. 69–84
9. V.S. Pai, M. Aron, G. Banga, M. Svendsen, P. Druschel, W. Zwaenepoel, E.M. Nahum, LocalityAware request distribution in cluster-based network servers, in *Proceedings of the 8th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS VIII)*, San Jose, CA, pp. 205–216 (1998)
10. A. Lakshman, P. Malik, Cassandra—a decentralized unstructured compute system, in *The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware* (Big Sky Resort, Montana, 2019)
11. I. Canadi, Integrating RocksDB with MongoDB, <http://rocksdb.org/blog/1967/integrating-rocksdb-withmongodb-2/> (2015)

Performance Comparison of Different Machine Learning Algorithms on Hindi News Classification



Monika Arora, Bhumika Dhingra, Dhruv Gupta, and Dajinder Singh

Abstract Text classification is a process of categorizing text into predefined classes. It plays an important role in extracting useful information which can be further used for sentiment analysis, spam detection, content tagging, etc. In today's world a large number of Hindi text documents are generated by government sites, various magazines, and newspapers which are required to be classified. Since limited work has been done on Hindi language, so, we tried to classify Hindi news text collected from newspapers like Jagran, Navbharat, Patrika, etc. using different machine learning algorithms. Since machine learning algorithms are not able to work directly on the extracted text, so we applied pre-processing and feature engineering techniques such as count vectorizer, TF-IDF and word2vec. Such pre-processing is challenging due to the presence of multisense words, conjunctions, punctuations and special characters. Our model accepts Hindi news headlines of different predefined categories such as Entertainment news, Sports News, Tech news and Lifestyle news. The corpus size containing unique words after pre-processing was 54,44,997 words. Out of different combinations, multinomial Naïve Bayes with count vectorizer outperformed with an accuracy of 85.47%.

1 Introduction

Text Classification is a process of labeling the text with predefined tags. Text classification can also be named as Text Tagging or Text Categorization [1]. It is one of the main activities involved Natural Language Processing. Text tagging broadly used in applications like Spam Filtration, Sentiment Analysis, Auto Tagging, Topic labeling, etc. classifying the textual content into predefined labels helps the users to navigate through the web data or any applications easily, as it helps in simplifying the content.

M. Arora (✉) · B. Dhingra · D. Gupta · D. Singh
IT Department, Bhagwan Parshuram Institute of Technology, New Delhi, India
e-mail: monikaarora@bpitindia.com

Text classification is basically used for making the things automated and simpler with minimum human intervention using Artificial Intelligence like in classification of books in libraries or segmentation of news in articles. In this paper, we have analyzed the impact of different machine learning algorithms on classification of Hindi news articles.

Hindi is one of the most spoken languages in India. It is also a recognized minority language in South Africa. A lot of textual data in Hindi language is generated on the web in the form of news, messages, and articles, which remains unclassified because not much work has been done on Hindi language. Therefore, this type of data is needed to be classified for further analysis like spam filtration, opinion mining, text mining, data mining, etc. [2]. Classification, in simple terms, is a process of categorization of data under different categories. Different machine learning algorithms are present which can be used to process the data which help in classification.

The purpose of this paper is to create a machine learning model for Hindi News Classification, in which the news is classified into four predefined categories, namely—Entertainment, Sports, Lifestyle, and Technology. The model created is trained on a large dataset with different machine learning algorithms (Logistic Regression, Multinomial Naïve Bayes, Passive Aggressive) out of which Multinomial Naïve with count vectorizer outperformed with an accuracy of 85.47% on testing data. Since data pre-processing is a crucial step, the removal of English Characters, numbers, special characters, noise, punctuations, unwanted spaces, stop words, insignificant news along with lemmatization is important.

This paper is divided into various sections. Section 2 contains Literature Review which gives a brief about the previous work done in the same domain, that is, Text Classification. Section 3 explains the steps followed in order to achieve the results. It starts from collecting the dataset, then collecting the corpus of Hindi words, data cleaning, data pre-processing, Feature Engineering, Feature Selection up to the process of Model Building. Section 4 contains the results we have obtained after testing the model on validation data and testing data. And, last Sect. 5 concludes the summary of the work as conclusion.

2 Literature Review

Hassan and Zaidi [3] worked on Urdu news Headline from BBC Urdu and Urdu Point which created a corpus containing 141289 words of 8 different categories. On comparison of different machine learning algorithms for classification of news categories, ridge classifier achieved maximum accuracy of 87%.

Bilal et al. [4] proposed three classification models for text classification using Waikato Environment for Knowledge Analysis (WEKA). The machine learning models were created on the opinion dataset which was extracted from the blog containing 150 positive opinions and 150 negative opinions. In this, the results were compared for KNN, Naïve Bayes and Decision Tree algorithms out of which Naïve Bayes outperformed in terms of accuracy, precision, recall and F-measure. Alam and

ul Hussain [5] address this problem and transform Roman-Urdu to Urdu transliteration into sequence-to-sequence learning problem. Roman-Urdu to Urdu corpora was created and pass it to neural machine translation model that predicts sentences up to length 10 while achieving BLEU score of 48.6 on the test set.

Usman et al. [6] applied five classification techniques (Linear SGD, Multinomial Naïve Bayes, Bernoulli Naïve Bayes, Linear SVC, Random forest and Max voting) on Urdu language corpus which consisted of 93400 features. After comparison the best results were given by majority voting as it had a precision and recall value of 94%. Nidhi and Gupta [7] worked on domain-based classification on Punjabi language using ontology-based classification, hybrid approach (combination of Naïve based and ontology-based classification), centroid-based classification, and Naïve Bayes classifier. According to the results, Ontology-based classification and hybrid approach provided an accuracy value of 85%.

Kadhim [8] surveyed on different term weighing methods along with comparison among different supervised machine learning classification techniques. This paper focuses on Naïve Bayes, SVM, and KNN and based on the comparison KNN with TF-IDF performs well. Kowsari et al. [1] discuss different text feature extractions and classification algorithms. Different feature extraction methods used in this paper are TF-IDF, word2vec, GloVe, and FastText combined with different text pre-processing methods like PCA, LDA, NMF, t-SNE, autoencoders. Different classification algorithms used on this pre-processed text are Logistic Regression, Naïve Bayes Classifier, KNN, SVM, Decision tree classifier, etc. Based on this, advantages and pitfalls of these algorithms on text classification use case were highlighted.

Rani and Satvika [9] use works on the categorization of text documents written in Indian Languages using KNN and i-KNN algorithm. This paper concludes that accuracy on Hindi and English language is better than the existing work done on Telugu, Tamil, and Kannada language. Moreover, based on the comparison results, i-KNN has an accuracy value of 0.99 for Hindi language which is much better than the accuracy value of KNN.

3 Methodology

This methodology contains a step-wise procedure; Starting with the collection of Hindi news articles for creating corpus and then pre-processing the data for cleaning and feature engineering purpose to apply different machine learning algorithms for classification of the news as per predefined classes. The process flow as depicted in Fig. 1.

A. Corpus Collection

Training of data is a very crucial step in the development of a model that uses supervised machine learning algorithm. For training purpose, the data was collected from random newspapers which was classified into different categories. This data consists of the news of all the four categories in which

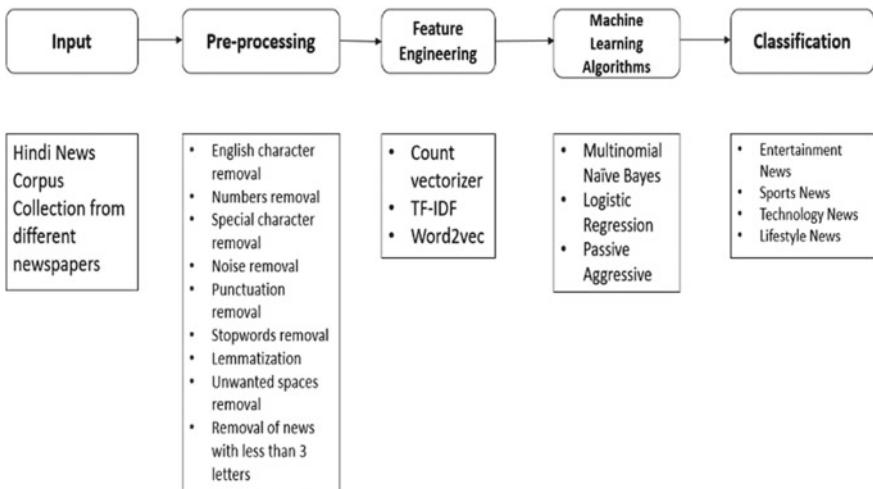


Fig. 1 Summarizes the process

the classification has to be done. The data collected was in textual format and contained a total of 54,44,997 words.

The four predefined classes were Entertainment news, Sports news, Technology news, and lifestyle news.

There were total of 99664 news of Entertainment category, 115867 news of sports category, 18692 news of Technology category and 58897 news of Lifestyle category.

The data classification legend is shown in Table 1.

B. Pre-processing

- i. **English character removal**—This process comprises removal of English characters present in the Hindi news text. The presence of English alphabets or characters in the news item was removed to get the pure Hindi text for feature engineering and training purpose.
- ii. **Numbers removal**—This process comprises removal of numbers present in the Hindi news text. There were some news headlines which have numbers present in it. These numbers were also removed as a step-in data cleaning process.

Table 1 Labels for different news classes

Classes of news	Labels
Entertainment news	1
Sports news	2
Technology news	3
Lifestyle news	4

- iii. **Special character removal**—This process comprises removal of special characters such as (!, +, -, =, <, >, \$, &, etc.). The presence of special characters does not play any role in the classification so it was preferred to remove these characters. These special characters consisted of exclamation marks, dollar (\$) symbol, and (&) symbol, etc.
- iv. **Noise removal**—This process comprises removal of some noisy elements present in the text such as ('\\", 'u0240d', etc.). These words generally arise due to decoding of short forms or symbols of different Hindi words. These short forms were also removed from the dataset to make it cleaner for training purpose.
- v. **Punctuation removal**—This includes removal of punctuation mark such as (<, >, %, !, -, :, ;, etc.).
- vi. **Stopwords removal**—Frequently occurring words which contain very less or no information are known as stopwords [10]. These words are not useful for the proposed classification models. We encountered 230 unique stopwords that should be removed from the dataset, resulting in extraction of meaningful corpus for the classifiers. Below are few examples of stopwords from the training dataset.
- vii. **Lemmatization**—This process comprises grouping of inflected forms of a word so as to perform analysis on single form of a word [11].
- viii. **Unwanted Spaces removal**—This process comprises removal of some extra spaces which were generated due to above pre-processing steps.
- ix. **Removal of news with less than three letters**—This process includes the removal of news items with less than or equal to 3 words as it is difficult to classify the news headline with just 3 words because these news rarely have the keywords on which the classification is done. These news headlines were considered as irrelevant news items and so were dropped from the dataset (Figs. 2 and 3).

- 1 नई दिल्ली (जेएनएन)।
- 2 स्मार्टफोन निर्माता कंपनी TCL ने अल्काटेल ब्रैंड के दो हैंडसेट्स भारत में पेश किए हैं।
- 3 Alcatel A5 LED और A7 को एकसकलूसिय तौर पर इ-कॉमर्स वेबसाइट अमेजन इंडिया से खरीदा जा सकेगा।
- 4 इसके अलावा जेन मोबाइल्स कंपनी ने Admire Unity स्मार्टफोन लॉच किया है।
- 5 इसकी कीमत 5,099 रुपये है।
- 6 जानें इनकी डिटेल्स: Alcatel A5 LED और A7 की कीमत: Alcatel A5 LED की कीमत 12,999 रुपये है।
- 7 लॉच ऑफ के तहत फोन के साथ 3100 एमएएच का पावर प्लस मोड बैटरी कवर दिया जाएगा।
- 8 इसकी कीमत 3,999 रुपये है।
- 9 वहीं, Alcatel A7 की कीमत 13,999 रुपये है।
- 10 लॉच ऑफ के तहत इस फोन के साथ 2,499 रुपये का मूवबैंड फिटनेस ट्रैकर, बन टाइम स्क्रीन रिप्लेसमेंट समेत 20 जीवी तक अतिरिक्त डाटा दिया जाएगा।

Fig. 2 Data before pre-processing

- 1 नई दिल्ली जेरएनएन
 2 स्मार्टफोन निर्माता कंपनी अल्काटेल ब्रैड हैंडसेट्स भारत पेश किए
 3 एक्सवलूसिव तौर ई कॉमर्स वेबसाइट अमेजन इंडिया खरीदा सकेगा
 4 अलवा जेन मोबाइल्स कंपनी स्मार्टफोन लॉन्च
 5 कीमत रुपये
 6 जानें इनकी डिटेल्स कीमत कीमत रुपये
 7 लॉन्च ऑफर तहत फोन एमएच पावर प्लस मोड बैटरी कवर जाएगा
 8 कीमत रुपये
 9 कीमत रुपये
 10 लॉन्च ऑफर तहत फोन रुपये मूवबैंड फिटनेस ट्रैकर वन टाइम स्क्रीन रिप्लेसमेंट समेत जीबी अतिरिक्त डाटा जाएगा

Fig. 3 Data after pre-processing

C. Feature Engineering

- I. **Count vectorizer:** Count vectorizer is used to create the vector of term or token count from the collection of text document in the form of a matrix based on the count of each feature [12]. Some pre-processing is also done by count vectorizer before the conversion of text into vector. In this case, after applying count vectorizer to the corpus, 80,093 unique words were found and their vectors were created.
- II. **TF-IDF vectorizer:** TF-IDF stands for Term Frequency-Inverse Document Frequency [12]. In this, the importance of a word is measured by calculating the term frequency and inverse document frequency for all unique words and then converting them into vectors. This technique was used to convert top 50,000 unique words into vectors.

$$tf_{t,d} \times idf_t \quad (1)$$

$$TF_{t,d} = (\text{frequency of a term } t) / (\text{length of the document}) \quad (2)$$

$$IDF_t = \log \left(\frac{\text{number of documents}}{\text{number of documents with term } t} \right) \quad (3)$$

- III. **Word2vec:** This algorithm uses a neural network to learn the associations among the words from the corpus of text. This type of feature engineering technique is based on two algorithms, namely, continuous bag of words (CBOW) and skip gram algorithm [13]. Word2vec represents distinct words as a list of numbers called as vector. In this case, average sentence length is 10 words per sentence.

D. Building Machine learning models

- 1. **Multinomial Naïve Bayes:**

Multinomial Naïve Bayes is a type of Naïve Bayes algorithm which generally undertakes the document as a collection of text and processes it by taking its frequency and its other information into account [14]. All Naïve Bayes algorithms work on the principle of Bayesian theorem and predict

the class of the given vector on the basis of probability and likelihood. In our case, the multinomial Naïve Bayes algorithm gave the accuracy of 85.47%, 40.01%, and 27.29% with feature engineering technique count vectorizer, TF-IDF, and word2vec, respectively.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (4)$$

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c) \quad (5)$$

2. Logistic regression:

It is one of the basic classification algorithms which can be defined as statistical model that creates a logistic function to model binary dependant values. Here binary dependents mean the representation in the form of “0” and “1”. The sigmoid function predicts the probability of the feature to be in a class. If the probability is greater than 0.5 then that feature belongs to class 1 otherwise class 0 [15]. It can also be used for multiclass classification by assuming one class as positive (1) and all other classes to be negative (0) and repeating same process with the negative class, i.e., again by dividing into positive class and negative class. For example, in our case logistic regression algorithm gave accuracies of 83.13%, 31.69%, and 27.40% with feature engineering techniques count vectorizer, TF-IDF, and word2vec, respectively.

$$\log\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m \quad (6)$$

3. Passive Aggressive:

Passive aggressive is online learning algorithm used for both regression and classification. It is an online algorithm which can be used on the large stream of data. It works on the principle of providing weight to the vector and if the dot product of the weight vector to the word vector is greater than +1 then the predicted class is same as actual class, i.e., the vector is passive and if the dot product comes out to be less than -1 then the predicted class is not same as actual class, i.e., the word vector is aggressive. And thus, it works for different optimized weight vector in different iterations for better predictions. For example, in our case logistic regression algorithm gave accuracies of 48.77%, 31.69%, and 27.40% with feature engineering techniques count vectorizer, TF-IDF, and word2vec, respectively.

$$\bar{W}_{t+1} = \bar{w}_t + \frac{\max(0, |y_t - \bar{w}^T \cdot \bar{x}_t| - \varepsilon)}{\|\bar{x}_t\|^2 + \frac{1}{2c}} \text{sign}(y_t - \bar{w}^T \cdot \bar{x}_t) \bar{x}_t \quad (7)$$

4 Implementation and Results

We applied pre-processing on large Hindi news dataset to get the relevant data and then we applied different feature engineering techniques, namely, Count vectorizer, TF-IDF, and Word2vec to find the best combination of different machine learning algorithm with feature engineering technique. We used 3 machine learning algorithms, i.e., Multinomial Naïve Bayes algorithm, Logistic Regression, and Passive Aggressive. We got different accuracies on using different combinations of feature engineering techniques and machine learning algorithms. The most effective combination of feature engineering technique along with machine learning algorithm was count vectorizer technique with multinomial Naïve Bayes algorithm getting accuracy of 85.47%.

The below matrix describes the accuracies and F1 score of different techniques with respective machine learning algorithms (Tables 2 and 3).

The model was tested on the set of completely different test dataset and the result came out to be impressive (Figs. 4, 5, and 6).

Table 2 Accuracy matrix

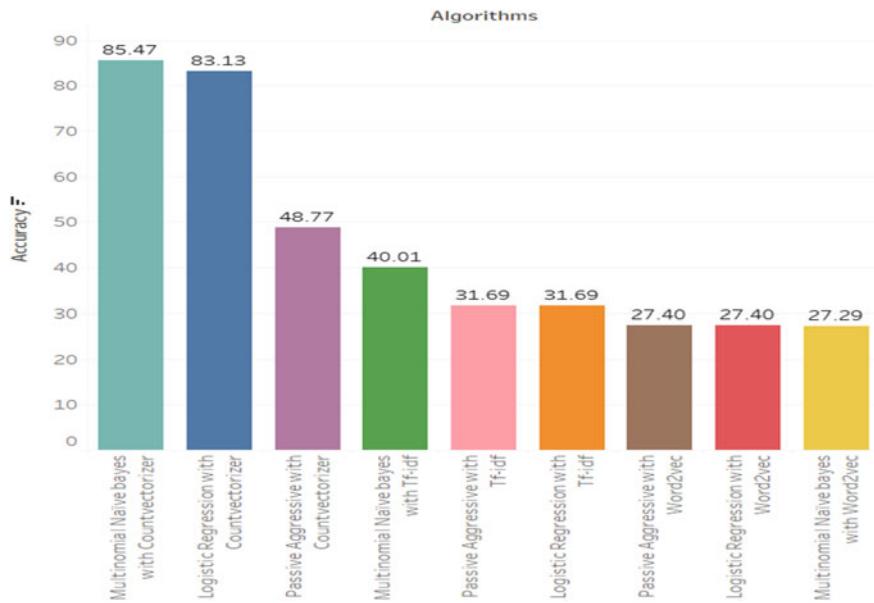
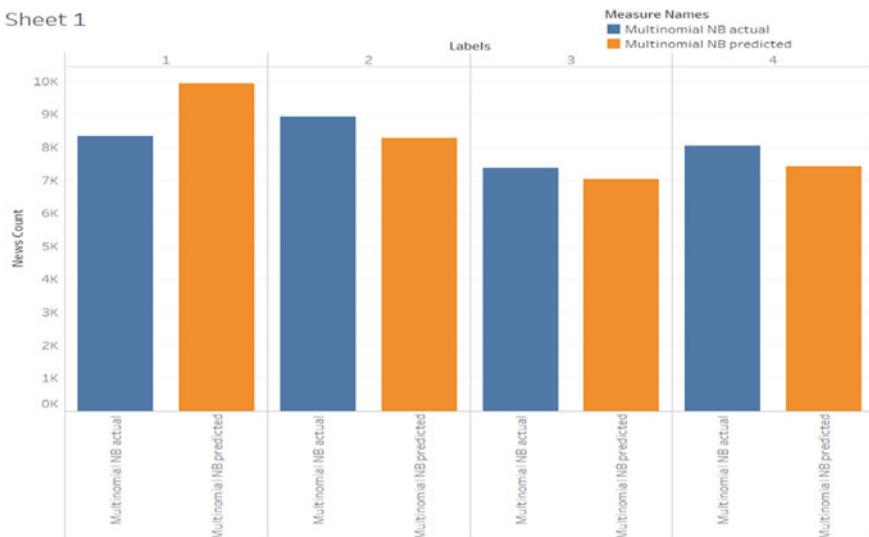
Feature selection technique versus machine learning algorithm	Multinomial Naïve Bayes	Logistic regression	Passive aggressive
Count vectorizer	85.47	83.13	48.77
TF-IDF	40.01	31.69	31.69
Word2Vec	27.29	27.40	27.40

Table 3 F1 score

Feature selection technique versus machine learning algorithms	Multinomial Naïve Bayes	Logistic regression	Passive Aggressive
Count vectorizer	0.95	0.90	0.62
TF-IDF	0.43	0.43	0.43
Word2vec	0.30	0.44	0.30

Unnamed: 0		News	Actual_Class	Predicted_Class
0	0	अभिनवी करीना कपूर सैफ अली खान बेटे तैमूर इत...	Entertainment	Entertainment
1	1	टीम इंडिया कप्तान बॉलीवुड एक्ट्रेस अनुष्का शर...	Sports	Sports
2	2	अमेरिकी निर्माताओं रोजगार सृजन नवोन्मेष देने अ...	Technology	Technology
3	3	खास वायु प्रदूषण बढ़ता	LifeStyle	LifeStyle

Fig. 4 Comparison of actual and predicted results of multinomial Naïve Bayes with count vectorizer

**Fig. 5** Comparison of different accuracy values**Fig. 6** Comparison between the news count for different labels (actual vs. predicted)

5 Conclusion

In this paper we have used a large dataset of Hindi news containing four different categories to create a machine learning model for Hindi news classification. We pre-processed the data to extract the relevant features and then applied three different feature engineering techniques—Count Vectorizer, TF-IDF, and Word2Vec, to make the data ready to be trained by machine learning algorithms. After that, we have applied three machine learning algorithms—Multinomial Naïve Bayes, logistic regression, and passive aggressive, on each feature engineering technique to find the best combination. As a result, Multinomial Naïve Bayes with count vectorizer outperformed with an accuracy of 85.47%.

These machine learning models have been tested on a different testing data (other than training data and validation data), and the multinomial Naïve Bayes with count vectorizer gave the most accurate predictions.

6 Future Scope

The performance can be enhanced by optimizing the pre-processing and feature engineering techniques. It is also possible to enhance the performance by creating different deep learning architectures or by creating a model on a larger dataset. There is a higher chance that deep learning will give better results than machine learning on Hindi news text classification. It has been planned to apply RNN, Bi-directional RNN, LSTM, and GRU architectures to the same pre-processed news with same feature engineering techniques. Moreover, a web application can also be integrated with such machine learning model to help organizations and institutions in better management of data. This project can also be helpful in creating more projects such auto-tagging, sentiment analysis, data mining, opinion mining, spam filtration, detection of fake news, etc.

References

1. K. Kowsari, K.M. Jafari, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text classification algorithms: a survey. *Information* **10**(4), 1–68 (2019)
2. A. Khan, B. Baharudin, L.H. Lee, K. Khan, A review of machine learning algorithms for text-documents classification. *J. Adv. Inf. Technol.* **1**, 4–20 (2010)
3. S.M. Hassan, A. Zaidi, Urdu/Hindi News Headline, Text Classification by Using Different Machine Learning Algorithms. <https://doi.org/10.13140/RG.2.2.12068.83846>. Accessed Dec 2018
4. M. Bilal, H. Israr, M. Shahid, A. Khan, Sentiment classification of Roman-Urdu opinions using Naïve Bayesian, Decision Tree and KNN classification techniques. *J. King Saud Univ. Comput. Inf. Sci.* **28**(3), 330–344 (2016)

5. M. Alam, S. ul Hussain, Sequence to sequence networks for Roman-Urdu to Urdu transliteration, in *International Multi-topic Conference (INMIC)*, Lahore (2017), pp. 1–7. <https://doi.org/10.1109/INMIC.2017.8289449>
6. M. Usman, Z. Shafique, S. Ayub, K. Malik, Urdu text classification using majority voting. *Int. J. Adv. Comput. Sci. Appl.* **7**, 265–273 (2016)
7. K. Nidhi, V. Gupta, *Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing (SANLP) COLING 2012*, Mumbai, pp. 109–122, Dec 2012
8. A.I. Kadhim, Survey on supervised machine learning techniques for automatic text classification. *Artif. Intell. Rev.* 1–20 (2019). <https://doi.org/10.1007/s10462-018-09677-1>
9. K. Rani, Satvika, Text categorization on multiple languages based on classification technique. *Int. J. Comput. Sci. Inf. Technol.* **7**(3), 1578–1581 (2016)
10. J. Kaur, P.K. Buttar, *Int. J. Future Revolut. Comput. Sci. Commun. Eng.* **4**(4), 207–210 (2018)
11. V. Balakrishnan, E. Lloyd-Yemoh, Stemming and lemmatization: a comparison of retrieval performances. *Lect. Notes Softw. Eng.* **2**, 262–267 (2014)
12. H. Arif, K. Munir, A.S. Danyal, A. Salman, M.M. Fraz, Sentiment analysis of Roman Urdu/Hindi using supervised methods, in *International Conference on Innovative Computing* (2016), pp. 1–7
13. B. Jang, I. Kim, J.W. Kim, Word2vec convolutional neural networks for classification of news articles and tweets. *PLoS One* (2019). <https://doi.org/10.1371/journal.pone.0220976>
14. G. Singh, B. Kumar, L. Gaur, A. Tyagi, Comparison between multinomial and Bernoulli Naïve Bayes for text classification, in *International Conference on Automation, Computational and Technology Management (ICACTM)*, London, United Kingdom (2019), pp. 593–596. <https://doi.org/10.1109/ICACTM.2019.8776800>
15. Y.R. Rochlani, A.B. Raut, P.D. Kaware, An overview of sentiment analysis. *J. Seybold Rep.* **15**(9), 163–167 (2020)

ConvLSTM for Human Activity Recognition



Ramendra Singla, Shubham Mittal, Alok Jain, and Deepak Gupta

Abstract The research in Human Activity Recognition (HAR) using wearable probes and pocket devices has intensified to further understand and inherently foresee human behavior and their intentions. The researchers are seeking a system to consume the least amount of allocated resources to identify the consumer's activity being performed. In this paper, we propose a state-of-the-art deep learning-based activity recognition architecture, a Convolutional Long Short-Term Memory (ConvLSTM) network . This ConvLSTM approach significantly improves the accuracy of classification of the six activities from raw data without the use of any major aspect of feature engineering hence, reducing the complexity of the model with a very minor pre-processing procedure. Our proposed model is able to achieve a staggering 94% accuracy on the UCI HAR public dataset. During performance comparisons with earlier models, we were able to notice profitable improvements against Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) Network, Deep Neural Network (DNN) models, and also against linear and non-linear machine learning models which heavily depend upon manually manufactured featured data.

Keywords Human Activity Recognition (HAR) · Convolutional Neural Network (CNN) · Long Short-Term Memory Network (LSTM) · Convolutional Long Short-Term Memory (ConvLSTM) · Deep learning · UCI HAR dataset · Time Series Classification (TSC) · Neural network · Magnetometers · Accelerometers · Gyroscopes · Sensors · Inertial Measurement Unit (IMU)

R. Singla · S. Mittal (✉) · A. Jain · D. Gupta
Maharaja Agrasen Institute Of Technology, Plot No 1 Rohini, Sector 22, PSP Area,
Delhi 110086, India
e-mail: deepakgupta@mait.ac.in

1 Introduction

Inertial Measurement Unit (IMU) has become the obvious choice for regularly measuring and monitoring activities by various equipment, machinery, animals and not to mention human beings. The classification problem of Human Activity Recognition (HAR) is built upon these IMU making it a prevalent research for many researchers across the globe. The Inertial Measurement Unit (IMU) has also made it very concrete to facilitate data privacy and user friendly. Some of the earlier researched and proposed approaches [10–12] make it a very strenuous task as it majorly requires a huge abundance of resources allocated and subject matter specialists, the major hurdles to name a few. The evolution of deep learning has made a major impact in many fields especially and including in the research area of Human Activity Recognition (HAR) and made our lives very simple. Hammerla et al. [4] claim that deep learning revolutionizes the biggest inclination in the machine and deep learning over the past couple of years. Frameworks such as Keras [1], TensorFlow (from Google), Scikit, and PyTorch (from Facebook) have helped to create and experiment on Machine and Deep Learning models.

Since 2012, the greatest of advances has happened using deep learning in many areas of research and practice Human Activity Recognition has not been paid much notice. The abundant usage of smart devices with inbuilt smart sensors such as accelerometer and gyroscope which constantly receives various devices data connected to the servers allowing the constant process of measuring the activity of the user. The latest research has also created varieties of designs particularly with the present-day keen gadgets having the ability to recognize and measure the activity by themselves. These devices have bigger storage memory, better processing power, and efficient sensors.

Just by the usage of raw sensor data, it is now possible to make fast and efficient trainable models with the evolution of deep learning. Machine Learning algorithms such as Support Vector Machines (SVMs) [11] and K-nearest Neighbors [10] which are in minimal usage for research purposes were recommended but required a lot of manual work to be processed including domain expertise, preparation, collection of data, and pre-processing that made it to be very difficult to operate although giving remarkable accuracies and improvements.

Deep Learning has given an amazing breakthrough that allows us to procure essential features and evidently make efficient predictions on its own provided we are able to set up a convenient architecture and design of the model to be trained and pre-process the input according to the architecture. The major deep learning algorithms published in HAR include Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) [3] which allows spatial depth and temporal deep nature, being very dense feed-forward neural networks. Each of these published ideas showcased that each of CNN and LSTM has its own pros and cons. To improve this further, the idea of CNN-LSTM [2] was put forward to make use of the strengths of CNN and LSTM together to give a better prediction and hence were successfully able to achieve the same.

This research has brought the idea of ConvLSTM [5] classifier to the picture for human activity recognition. The idea behind ConvLSTM is very simple, and it is a leap forward from the original idea of LSTM. In the original LSTM network, we had a feature value that was passed and multiplied, but that initial step is replaced with convolutional operation with the input of the image matrix passed. Background and the research in previous work will be discussed in Sect. 2. The dataset used will be discussed in Sect. 3. The implementation and architectural setup will be discussed in Sect. 4. The accuracy and improvements in comparison with the previous models on the UCI HAR dataset are described Sect. 5. In Sect. 6, we will have our conclusions along with the future work that can hold for this field.

2 Background and Previous Work

A large amount of research has taken into account human activity recognition using a series of ideas and breakthroughs as shown by Anguita et al. [11], Eyoobu and Han [12], and many more. We are going to discuss the basis of the deep learning-based human activity recognition connecting the previous research to the one proposed by us and also differentiating how our differs from others.

2.1 Data Collection for Human Activity Recognition

Magnetometers, accelerometers, and gyroscopes are some of the sensors that are built into the IMU devices and the latest gadgets are being used to identify the activity being performed by the user, this is what the current HAR problem aims to achieve efficiently [6]. The different signals from the above-mentioned sensors are collected on a regular time interval hence classifying this problem to be a Time Series Classification (TSC).

Ismail Fawaz et al. [7] proposed the definition of a time series classification is when a time series such as $Z = [z_1, z_2, \dots, z_T]$ is an ordered (with time) sets of real values and each set has the length as T . Further, the dataset is defined as $D = (Z_1, y_1), (Z_2, y_2), (Z_3, y_3), \dots, (Z_N, y_N)$ where Z_i is a series of time data with y_i as the class label for Z_i . The base for a TSC problem using the machine and deep learning is based on the probability distribution to classify y_i for their corresponding Z_i in the dataset D . Many machine and deep learning algorithms have been suggested to solve the HAR problem being a TSC such as SVM by Anguita et al. [11], LSTM [12], CNN [3, 7].

2.2 HAR with Deep Learning

Since 2012, Alexnet [3] have emerged as one the ImageNet model and it is also the first Deep Neural network model with a huge margin of accuracy improvement. This created the numerous applications of deep neural networks in vast areas of domain. To name a few Natural Language Processing (NLP), Computer Vision (CV), etc. This leaded many researchers to try this on creating promising results in HAR using deep learning [3, 9, 11, 12]. Deep learning gives us the liberty of not investing time and effort for engineered features, and hence the expert knowledge of this domain is not essential.

The machine learning method of modeling the HAR problem is feature engineering from x-, y-, z-axis raw data from the sensors as mentioned earlier, which is further divided into multiple samples in the form of windows and statistically analyzed in frequency and time domain [11]. The features extracted from these are trained and predicted, producing excellent results but is very dependant on many basic factors such as the gadget used to collect the results, the conditions around the readings were collected such as the weather, the humidity as the sensors may capture different results for different situations. Therefore, the smallest change in the conditions could develop different results as the model is not able to capture the change in signals and rather the values themselves. This presents be a huge challenge and not acceptable from a practical standpoint. The environment issue is resolved by the usage of DL which looks at the change of signals making it a much more practical choice.

2.3 Convolution Neural Networks and Long Short-Term Memory Networks for HAR

CNN has become a popular and efficient choice in many areas of study holding its property very similar to the visual cortex of our brain, hence has found a great deal of usage with images coming into play and image recognition problems [8]. Hammerla et al. [4] proposed that CNN was able to surplus the accuracies in many fields including HAR due to their dimensionally deep nature. We will be making use of this nature in our model to our use.

The time-dependent nature of HAR must be taken advantage of as it helps to find the trend in change of signals every time as proposed by Hammerla et al. [4] to capture the movement of the user by the gyroscope and accelerometer. The adoption of the time-based nature of LSTM design keeping short- and long-term memory helps us finding effectively the change in signals over time and creating the features on their own. This attribute of LSTM has created astonishing results in various fields and is practically being used in multiple scenarios.

Eyobu and Han [12] incorporated the LSTM architecture in their modeling considering the temporal nature of LSTM using the engineered features rather than using the raw data collected. Kim et al. [9] and Mutegeki and Han [2] exploited the CNN

and LSTM hybrid architectures. Although [9] had their performances in terms of accuracies and error rate very similar to earlier proposed ML and DL models such as LSTM and SVM, etc., the [2] were able to improve their performances significantly by using different architecture altogether. [9] incorporated 2 LSTM layers and 3 CNN layers, and the LSTM layer having 128 units, whereas [2] used 1D convolutional layer, 1D maxpool layer, LSTM layer all wrapped in TimeDistributed followed by fully connected layer. We have a very different architecture altogether exploiting ConvLSTM rather than using CNN and LSTM layers separately.

3 Dataset

For the research of human activity recognition, we used the UCI HAR dataset as being used by many researchers proposed earlier, and hence to continue the uniformity, we proceeded with this widely used dataset. Dataset version 1 from [11] was primarily used from the gadget collected UCI HAR dataset. The University of California Irvine human activity recognition has been used in many other previous studies as it creates a standard level of comparison [2, 6, 10–12]. The dataset is used for finding the performance of transfer learning in [6] and the performance of CNN-LSTM-Dense on this specific dataset by Ronald Mutegeki and Dong Seog Han. Feature Engineering techniques using ML on the dataset by Jain and Kanhangad [10]. The suggested performance of feature engineering and augmentation techniques was proposed by Eyobu and Han [12].

The University of California Irvine dataset is based on a pool of 30 people age between 19 and 48 of age. The people were strapped to different sensors (gyroscope and accelerometer) to collect triaxial 3D signals [11]. Walking, Walking Upstairs, Walking Downstairs, Sitting, Standing, and Laying were the activities performed by each of the 30 people to have a completely balanced dataset. Butterworth filter was used to filter gravity and body acceleration to gather the sensor acceleration signals as well as noise filtering was used sampling the fixed bandwidth of 128 steps as described in [11]. 7352 train and 2947 test samples were used in the complete dataset for our modeling.

The model consisted of total of 9 inputs including linear and total accelerations as well as angular velocity with triaxial inputs of each representing x-, y-, and z-axis summing to be 9 raw inputs. The dataset was normalized and was directly fed to the built architecture which will be discussed in Sect. 4.

Figure 1 shows the snippet of the dataset used for our experimentation from the UCI HAR Dataset after normalization.

	ax	ay	az	wx	...	angleX	angleY	angleZ
0	0.2188	-0.6841	1.7085	8.5449	...	-34.3927	-8.1683	-1.2854
1	0.2300	-0.6777	1.7251	0.1831	...	-34.3652	-8.0035	-1.4777
2	0.2358	-0.7075	1.6978	-9.2773	...	-34.4312	-7.8662	-1.6589
3	0.2378	-0.7476	1.6519	-12.8784	...	-34.5355	-7.7454	-1.8237
4	0.2227	-0.7661	1.6191	-9.6436	...	-34.6124	-7.6245	-1.9830

Fig. 1 Snippet of dataset

4 Implementation and Architectural Setup

Figure 2 below shows the flow diagram of our suggested ConvLSTM model identifying and successfully predicting the 6 class labels. Keras API has enabled many researchers and practitioners to implement and review their idea with very minimal effort making all the major and very vastly used architectures readily available and could be straightforwardly imported and ready to use as proposed by Chollet [1].

The experiment was conducted in a controlled environment; we chose 1 2D ConvLSTM layer (tf. keras. layers. ConvLSTM2D) with 100 filters and kernel size = (1,5) which showcases the length of the convolution kernel. This layer used a ReLU activation function and leaving the other hyperparameters to be the default. Next, we suspected that this layer had too many parameters and hence could lead to overfitting hence we used a Dropout layer next with $p = 0.7$. The ConvLSTM layer experimented with many filter sizes and kernel sizes. Initially, the kernel size was kept constant at (1,3), but we later suspected that the model needed more learning and hence tried bigger kernels such as (1,7) and (1,9) too but (1,5) fitted the best. Similarly, we also changed the complexity of the model by using different filter sizes, but 100 sizes fitted the best. We would also like to mention that ConvLSTM also inherently has Batch Normalization to avoid exploding and vanishing gradient problem which is massive in scale for LSTM layers and hence; it is expected to be at a higher scale for a ConvLSTM cell. A single cell in ConvLSTM exploits the spatial depth of a convolution layer as well as the temporal nature of the LSTM layer making a very favorable choice for us. The input signal in the form of (None,128,9) was simply passed through the model without any changes.

Following the dropout layer, we passed the output from the dropout to the flatten layer. This flattens layer simply makes the output from the previous layer in linear form to be able to pass through the dense layers mapping the multi-dimensional layer

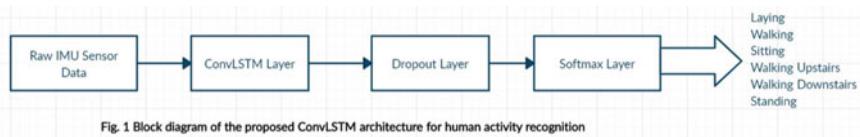


Fig. 1 Block diagram of the proposed ConvLSTM architecture for human activity recognition

Fig. 2 Flow Diagram of proposed ConvLSTM architecture

to our further linear layer. The output from ConvLSTM layer outputs (None, 1, 28, 100) and outputs through the flatten layer to be (None, 2800) feature maps. Then after the flatten layer, a Fully Connected (FC) of 100 neurons was connected in the model.

The output from the dense layer is then passed to a Fully Connected (FC) output layer with a softmax activation which classifies the given input into the 6 classes of the UCI HAR dataset.

For comparing our results with existing state-of-the-art architecture, we modeled the architecture similar to [2] naming it as CNN LSTM with 1 d Convolution layers and LSTM layers efficiently compare on the UCI HAR dataset. The hyperparameters were tuned with different batch sizes and learning rates and different optimizers, but very less impact was observed by changing the above mentioned, and hence, a lot of parameters were set to default. We used Adam optimizer with a learning rate of 0.01 with a mini-batch size of 64. We also tried changing the Learning rate based on the change of validation loss using Reduced LR Plateau but that didn't make too much difference too. We used around 400 epochs to train the model.

5 Results and Discussion

Figure 3 shown below the comparison of CNN LSTM with ConvLSTM on the UCI HAR dataset having a 6-class classification.

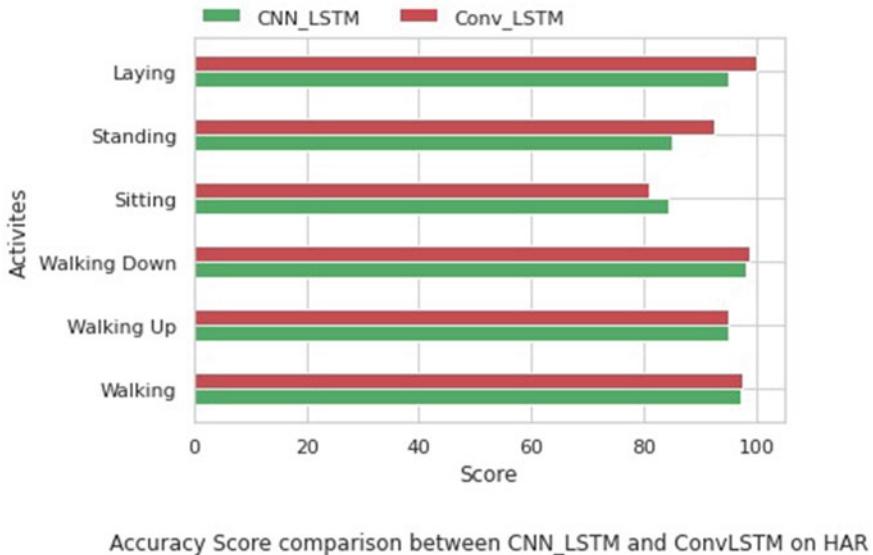


Fig. 3 Bar chart comparing CNN_LSTM with ConvLSTM on the 6 activities

After several experiments on the UCI HAR dataset with our ConvLSTM model with the state-of-the-art CNN LSTM model, we were able to conclude and prove that our model was able to perform significantly better with a huge margin of 2% in the overall performance. The results of this model along with the comparison are explained below.

We can clearly observe that ConvLSTM does a much better job in predicting activities with an overall performance of 94.10% in comparison to CNN LSTM having an overall performance to be 92.13%, hence a staggering difference of 2%.

We observe that ConvLSTM predicts with appreciably better accuracy in predicting activities such as Laying and Standing and a good amount of improvement for Walking Down and Walking. The other activities could be seen to be predicting the same or slightly less than CNN LSTM.

Figure 4 shows the confusion matrix that we were able to achieve by using the ConvLSTM.

The confusion matrix clearly shows that the model is successfully able to predict Walking Down and Laying without any problem with a 100% accuracy, and we also observe that Walking and Walking Up also have an appreciable accuracy of above 95%. The main drawback and future scope of improvement that we notice is in the confusion between Sitting and Standing, which is heavily causing the accuracy of Sitting. We see that both these activities have an only change of sensor signals only

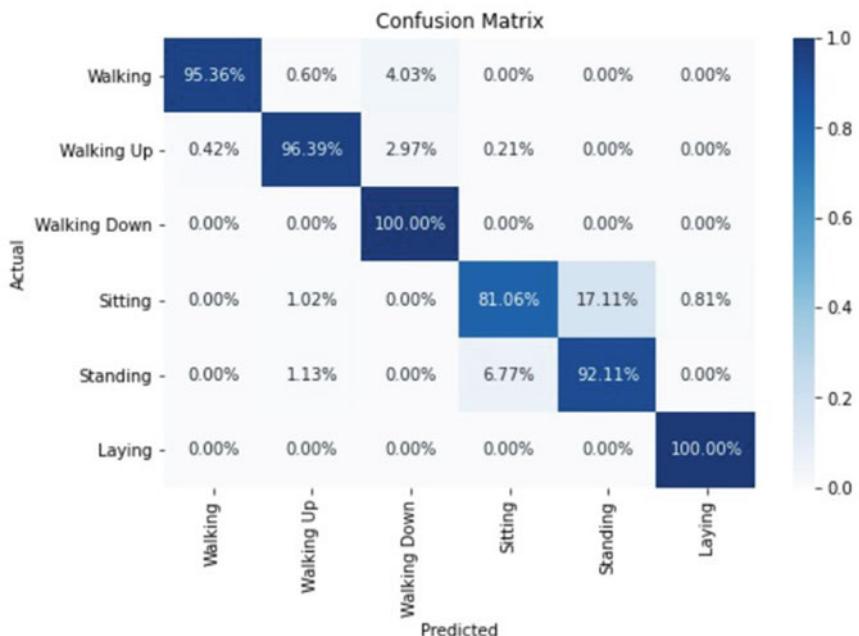


Fig. 4 Confusion matrix for ConvLSTM

for a very short period of time and stay constant after that hence causing the confusion between both these predictions. The future work is discussed in Sect. 6 along with the conclusion.

6 Conclusion and Future Scope

The paper proposed handling ConvLSTM addresses a huge impact in the improvement in comparison to the existing models and having a convolution layer incorporated in the LSTM layer in the same cell makes it a very promising layer because the layer can hold historic information, not just straightforward features but also the raw convolution image features. As discussed in [2], the main practical uses for ConvLSTM are video-based image moving tasks but we could see that a 1D kernel ConvLSTM has helped in HAR prediction. We also found that the model was able to train at a good speed of 27–28 ms per epoch, also giving a significant improvement in accuracy of 2% in the overall performance.

There is an immense future work that can be experimented with further. We could definitely test on more publicly available datasets to see the improvement against the existing state-of-the-art models, to gain more confidence in the ConvLSTM layer in the HAR area. The model could experiment with a more complex architecture with the ConvLSTM layer. As discussed in the results and discussion, we were able to determine that the model was successfully able to classify for continuous activities such as Walking, and Walking Down but unfortunately found it a bit difficult to classify Standing and Sitting which could be taken into consideration experimenting in the future.

References

1. H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.-A. Muller, Transfer learning for time series classification. *IEEE Int. Conf. Big Data* (2018)
2. R. Mutegeki, D.S. Han, ACNN-LSTM approach to human activity recognition. *IEEE Int. Conf.* (2020)
3. F.M. Rueda, R. Grzeszick, G.A. Fink, S. Feldhorst, M. Hompel, Convolutional neural networks for human activity recognition using Body-Worn sensors. *Informatics* **5**(2), 26 (2018)
4. N.Y. Hammerla, S. Halloran, T. Ploetz. Deep, convolutional, and recurrent models for human activity recognition using wearables (2016), [arXiv:1604.08880](https://arxiv.org/abs/1604.08880)
5. X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-C. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting
6. R. Mutegeki, D.S. Han, Feature-representation transfer learning for human activity recognition. *The 10th International Conference on ICT Convergence*
7. H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, P.-A. Muller, Transfer learning for time series classification. *IEEE Int. Conf. Big Data*, 1367–1376 (2018)
8. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **25**(2), 1097–110 (2012)

9. K. Kim, S. Choi, M. Chae, H. Park, J. Lee, J. Park, A deep learning based approach to recognizing accompanying status of smartphone users using multimodal data. *J. Intell. Inf. Syst.* **25**(1), 163–177 (2019)
10. A. Jain, V. Kanhangad, Human activity classification in smartphones using accelerometer and gyroscope sensors. *IEEE Sens. J.* **18**(3), 1169–1177 (2018). <https://doi.org/10.1109/JSEN.2017.2782492>
11. D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, A public domain dataset for human activity recognition using smartphones, in *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges, Belgium, 24–26 April (2013)
12. S.O. Eyobu, D.S. Han, Feature representation and data augmentation for human activity classification based on wearable IMU sensor data using a deep LSTM neural network. *Sensors* **18**, 2892 (2018)
13. S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
14. F. Chollet, Layer wrappers, Keras Documentation (2015), <https://keras.io/layers/wrappers/#timedistributed>. Accessed 1 Dec. 2019
15. R. Gómez, Understanding categorical cross-entropy loss, binary cross-entropy loss, Softmax loss, logistic loss, focal loss and all those confusing names (2019), https://gombru.github.io/2018/05/23/cross_entropy_loss/. Accessed 1 Dec. 2019

Mathematical Scanner (M-Scan) Mobile Application for Solving Simple Math Equations



Mamta Mittal, Gopi Battineni, Waqar Ahmad, Nitin Kumar, and Ravi Upreti

Abstract Some math's expressions are challenging to type in the calculator. As of this, this paper presents the necessity of developing math scanners that integrated with optical character recognition (OCR) technology for solving mathematical equations quickly and help students to cross-check their solution. The mathematical equation scanner proposed in this work can help to solve simple math equations in an efficient and timely manner. Scanners with OCR engines have been developed for different types of text recognition, and it became an interesting topic for many years. The developed application called M-Scan is an Android mobile application that is integrated with a camera. This camera simply captures the mathematical equations and solves them in a limited time. In this way, this device not only saves time but also increases accuracy with no errors.

Keywords M-scanner · Android application · OCR · Blob detection · Computer Algebra System (CAS)

1 Introduction

In 1917, Mary Jameson proposed the idea of the Octophone. The idea behind it is a free musical chord with setting up light from printed pages with the reflection of photoelectric cells [1]. It had used to be a great device for blind people by scanning the printed page at a speed of one word per minute. Thereafter, David Shepard developed another machine called ‘Gismo’ in 1951, which recognizes the 26 letters in Latin alphabets. It was used as a traditional typewriter and later evolved as a Farrington machine.

M. Mittal · W. Ahmad · N. Kumar · R. Upreti
Department of CSE, G.B. Pant Govt. Engineering College, Delhi, India

G. Battineni (✉)
Medical Informatics Centre, School of Medicinal and Health Products Sciences, University of Camerino, Camerino, Italy
e-mail: gopi.battineni@unicam.it

In the early 1960s, optical character recognition (OCR) technology was ready to establish as sorting of group mails in the US mail services. Later, Kurzweil technologies were released the initial Omni-font OCR system called CCD flatbed scanner [2]. After that, the Newton writing pad was launched in 1992 to recognize human handwriting. This device opens new doors of technology for conducting more developments in OCR. Later in 2006, Google adopted the OCR technology that fastened image scanners by involving neural networks. Because of this, neural networks allow OCR for self-pattern recognition. The contemporary advancement in OCR technology that happened in these years created the ability of letter or number recognition of any language even from miswriting documents [3].

Machines are not intelligent enough to understand the image data. Therefore, OCR is a piece of computer code implemented by machines unlike the human brain, which can simply acknowledge the text or character from a picture [4]. It can also complicate to perform as it is having different fonts, styles, and compound rules of languages. For computer science, techniques such as pattern classification, blob detection, image processing, and image segmentation are engaged to address different challenges. It notifies the user of the historical view, implementation, obstacles, and OCR approaches.

Despite it, Tesseract is an OCR engine for a variety of operating systems (OS), and it is free software delivered under the Apache license. It was primarily implemented by Hewlett-Packard (HP) as proprietary software in the 1980s, it was disclosed as open-source software in 2005 and further development has been done under Google since 2006 [5]. It was among the top three OCR engines in terms of character recognition accuracy in 1995. It is out there for UNIX OS, Windows, and waterproof OS X. However, due to restricted resources, it is only strictly tested by the developer's underneath Windows and Ubuntu [6].

There exists a limit of the scope of the tasks only to solve simple arithmetic expressions and systems of the linear equation up to one variable [7]. After getting the equation in the string form from OCR as output, the second phase of this research starts where we need to find out the solution for a particular equation using a computer algebra system, which is quite beneficial for solving the math-related tasks easily. The tool proposed in this work is in line with a computer algebra system (CAS), which is a mathematical software to handle mathematical expressions similar to the traditional calculations [8]. The event of the CAS within the last half of the twentieth century has become a piece of computer algebra or symbolic computation. It has several tasks of algorithms based on arithmetical equations such as polynomials.

This paper presents the development of an Android application that connects the gap between the technology of ancient pen and paper-based approach [9]. In this, the user will capture an image of an algebraic expression of printed fonts. Discussion on solution finding through simple expressions and linear equations with the help of some data structures were presented. The rest of the paper has structured as follows: Section 2 presents the methods and materials that were applied to develop the M-Scan application, Section 3 presents the results part including discussion, and Section 4 ends up with a conclusion and future developments.

2 Methods and Materials

The goal of the project is to build an Android-based application that captures an equation image to solve and displays the solution on the mobile itself. The basic method pipeline involved in solving an equation has presented in Figure 1.

2.1 Image capturing

Within the Android application, the user has two icons such as ‘select image’ and ‘show results’. The first ‘select image’ icon captures the image that is already saved within the memory of the mobile. However, the user must make sure about uploading the quality image (with high resolution) into the application.

2.2 Grey Scaling

Grey scaling is the method of converting a continual tone image to an image that a computer system can manipulate. The reason for distinguishing these images from the other types of colour images is that less amount of data needs to be provided for each pixel. Grey colour is the one that has the equal intensity of all the three elements such as red, green, and blue (RGB) space, and that is the only necessity to frame a single potency value for every pixel, as opposed to the three different intensities required to designate every pixel in an entire colour image [10]. Frequently, the grayscale

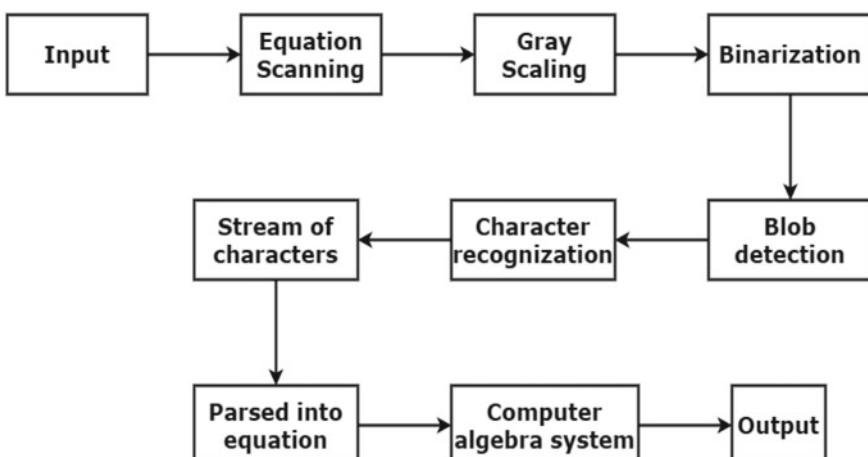


Fig. 1 The basic operational pipeline of M-Scanner

effectiveness is stored as an integer of eight bits having 256 possible spectres of grey from white to black. It is observed that the distinction between successive grey levels is remarkably higher when compared with the grey levels of the resolving power of the human eye. A lot of today's hardware that is used for display and image capturing only supports eight-bit images.

2.3 *Binarization*

In the process of image binarization, a threshold value is selected, and all pixels with values lower than this threshold are classified as black and others as white. The problem then is the way to choose the proper thresholding approaches [11]. A colour image comprises three channels of RGB with values from 0 to 255. The main objective behind the binarization is to convert the grayscale images into white and black (0 and 1) formats. Furthermore, binarization generates simple and sharp lines of diverse objects existing in the image. This feature extraction enhances AI model learning [10].

2.4 *Blob detection*

A binary large object (blob) refers to the connected component of all the pixels inside the whole binary image [12]. The term 'large' emphasizes the object of a certain magnitude, and the other term 'small' emphasizes the objects with noise. Blob analysis consists of three processes that explained below as shown in Figure 2.

Extraction: Blob extraction suggests that to separate the Blobs (objects) in a binary image. A Blob carries a bunch of associated pixels and we can decide if two pixels are connected or not by seeing the number of the connected component. There are two forms of properties like 4- and 8-connectivity, and the 4-connectivity is way lower than the 8-connectivity.

Characterization: The process to change the Blob into some characterization of numbers is known as the Blob characterization. Characterization of Blobs follows

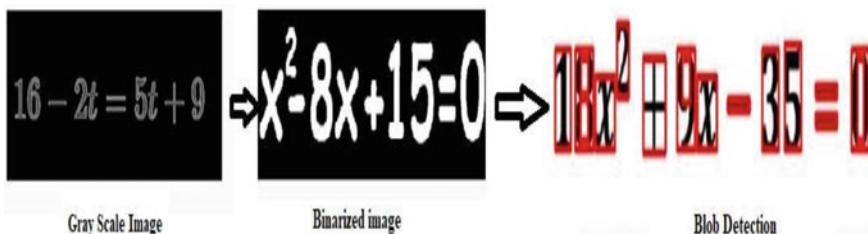


Fig. 2 Blob detection procedure

by an extraction step. In the Blob characterization method, two steps are followed as every Blob is represented by many characteristics in the first step and to apply some pattern matching approaches that can correlate the attributes of every Blob.

Categorization: In this phase, we decide the Blob type for checking as to whether the given Blob is a circle or not. But it is not easy to decide which Blobs are circle just only on their specific attributes [13].

2.5 *Character Recognition*

In most of today's applications, Tesseract OCR is used for text detection. Tesseract OCR engine offers the stream of character as output. OCR is the mechanical or electronic transformation of images of handwritten or typed, into the machine-encoded text, whether from a photo of a document, scanned document, a scene photo, or from subtitle text on an image (e.g., TV Telecast) [11, 14].

In the initial phase of the project, wolfram alpha was used to process the equation and evaluate the result. This method of solving the equation was online because a user must be dependent on the internet, also time-consuming. To overcome this, computer scientists decided to develop a system to compute the result, which was offline and fast as compared with the first method. In the second method, two types of equations (i.e., zero and one degree) are being solved by CAS.

2.6 *CAS*

CAS is an arithmetical software system having the capacity to govern the arithmetical expressions in a very way related to the standard calculations of mathematics. Specialized CAS has dedicated to some particular portion of arithmetic, like range theory, simple arithmetic.

- The general purpose of CAS focuses to be beneficial for a user operating in any field that desires regulation of arithmetic equations.
- A UI allows a user to feed the input and shows various arithmetical formulas.

2.7 *Explanation of the adopted approach*

2.7.1 *Simple Expression*

An expression is in the String form. The expression can have parentheses; it can be assumed that the parentheses are well matched as well as well placed. For simplicity,

it can be assumed that the only binary operations permitted are addition (+), multiplication (*), and subtraction (-). All the arithmetic expressions should be written in infix form (an operator should be placed in between the two operands).

2.7.2 The Algorithm Used to Solve the Expression

While not to the end of the expression string,

- Get the succeeding character of expression.
- If these characters, we have
 - An operand: put into operand stack.
 - Variable: get constant-coefficient, put into operand stack.
 - An opening parenthesis: put into operator stack.
 - A closing parenthesis.

While the operator on top of the operator stack is not an opening parenthesis,

- Exclude the operator from the operator stack.
- Exclude the two operands from the operator stack.
- Apply the operator on the two operands.
- Put the result back into the operand stack.
- Exclude the opening parenthesis from the operator stack.

While the operator stack is not empty, and the operator on the top of the operator stack has the similar or higher precedence as the current operator

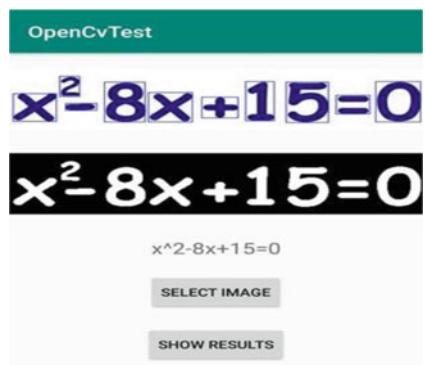
- Exclude the operator from the operator stack.
- Exclude the two operands from the operator stack.
- Apply the operator on the two operands.
- Put the result back into the operand stack.
- Push this operand onto the operator stack.

At this condition, the operator stack ought to be empty, and the operand stack ought to have just a single value in it, which is the value of the given expression [15]. The main work of the algorithm is to calculate the value of the variable used in each linear equation. Mathematical expressions can only contain binary operators like '+', '−', the variable, and its constant-coefficient. For no solution of an equation, it will return as 'No solution', for many solutions, return it as 'Infinite solutions', and for only a single solution, it will return a single integer.

The algorithmic rule is based on a two-pointer methodology. The main idea is to use two pointers method to change/update the value of those parameters: total sum and constant-coefficient value. On the LHS and RHS of '=', use opposite signs for each number to take care of a signed variable, which will flip when '=' is seen.

Now, just in the case of a single solution, the ratio of the effective result and constant-coefficient gives the desired result. In the case of infinitely many solutions,

Fig. 3 The starting screen of the application has two buttons. After the image is selected, blob detection and binarization of the image are shown one after another



both the effective result and constant-coefficient turn out to be 0. In the case of no solution, the constant of x turns out to be 0, but the effective result is non-zero [16].

3 Results

The sample screenshot of the developed scanner has presented in Fig. 3. The variables and constants are separated from this image and are grouped in the form of a string. The equation in the string form has been used to find the solution of the equation using different approaches based on the highest degree variable of the particular equation using the computer algebra system algorithms like to find the solution of the zero degree equation, first conversion of the equation into the infix to postfix notation takes place and then the stack is used to find the solution, and in case of one-degree equation, two pointers' approach is used that is described well in Section 3.

In Figure 4, the image of the simple expression has been taken and after that, the corresponding equation is converted into the binary form (i.e., 0 (black) and 1 (white)) and the equation in the form of a string and after that get the result by using stack. The image of the one-degree linear equation has been taken and the equation was converted into the corresponding string form. Based on that, we calculated the equation outcome using the corresponding algorithm and by employing expression evaluation.

4 Discussion

In this paper, an Android-based application for capturing the equation from an image containing a mathematical expression, and the method of finding the solution is being discussed. The pipeline of the whole application is also presented.

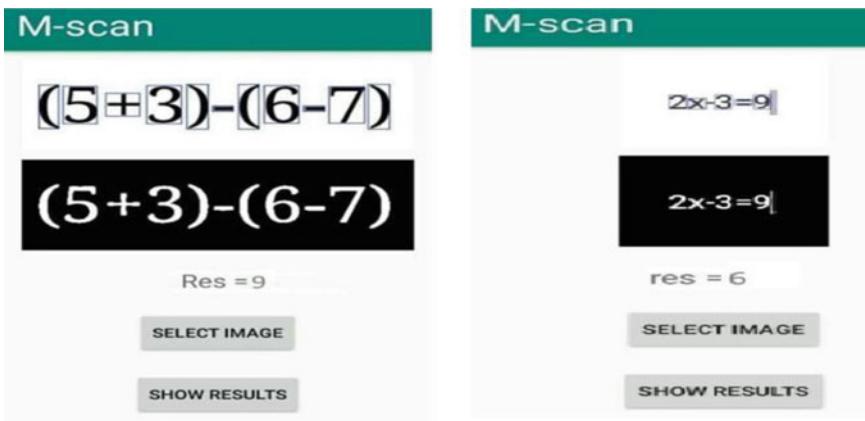


Fig. 4 Results of simple expression (left) and a one-degree equation (right)

Character recognition is not a new problem, but its source can be traced back to the systems before the origin of computer machines. The primary OCR systems were hardware devices that can perceive the characters, but those had very slow speed and low accuracy. Most people are gone through simple calculations during their daily lives. For students solving mathematical equations are an integral part of their education. To solve these equations, one must use a calculator or computer with a special format entry. To do that, OCR is employed to convert detected text into characters.

Various algorithms were used by the proposed system to tackle the problem statement. Big data processing and clustering of data is the need of the hour so various methods for the same have available in the literature [17, 18]. The image taken from the camera will act as the input to the system. A grayscale algorithm is applied to the image to convert it into a grayscale image. Then the grayscale image will be passed to calculate its intensity, after that, the image segmentation algorithm has been applied to the segmented image. To detect the stroke edges of the words, the post-processing algorithm has been applied and a binarized image will be generated.

The developed system consists of two components—an Android app for capturing the image of the equation from the printed fonts, and for displaying the recognized text and solution, and a server for performing the image process, text recognition, and equation solving algorithms.

The camera image of a mathematical expression is sent for pre-processing where it is processed from various phases such as feature extraction, segmentation, classification, pre-processing, and image acquisition and post-processing. Every aspect of the application, i.e., from image acquisition to displaying of the result is documented properly. For printed equation, text recognition is performed with the help of an OCR engine named Tesseract. The recognized equation is also displayed on the Android device for confirmation from the user. Then, the equation is solved with a computer algebra system, and then the solution is displayed on the Android device.

Conflicts of Interest No author was present any conflicts of interest during publication.

References

1. On a type-reading optophone. Proc. R. Soc. London. Ser. A, Contain. Pap. a Math. Phys. Character (1914), <https://doi.org/10.1098/rspa.1914.0061>
2. What is OCR (optical character recognition)?—Definition from WhatIs.com. <https://searchcontentmanagement.techtarget.com/definition/OCR-optical-character-recognition>. Accessed 09 Sep. 2020
3. S. Haq, Electronic invoicing gains as adoption barriers fall. Financ. Exec. (2007)
4. A. Chaudhuri, K. Mandaviya, P. Badelia, S.K. Ghosh, Optical character recognition systems, in *Studies in Fuzziness and Soft Computing* (2017)
5. C. Patel, A. Patel, D. Patel, Optical character recognition by open source OCR tool tesseract: a case study. Int. J. Comput. Appl. (2012). <https://doi.org/10.5120/8794-2784>
6. R. Smith, An overview of the tesseract OCR engine (2007), <https://doi.org/10.1109/ICDAR.2007.4376991>
7. E. Bhatia, Optical character recognition techniques: a review. Int. J. Adv. Res. Comput. Sci. Softw. Eng. (2014)
8. B. Kramarski, C. Hirsch, Using computer algebra systems in mathematical classrooms. J. Comput. Assist. Learn. (2003). <https://doi.org/10.1046/j.0266-4909.2003.00004.x>
9. A. Sikka, B. Wu, Camera-based equation solver for android devices. Ee368 (2012)
10. S.H. Shaikh, A.K. Maiti, N. Chaki, A new image binarization method using iterative partitioning. Mach. Vis. Appl. (2013). <https://doi.org/10.1007/s00138-011-0402-4>
11. D.P. Tian, A review on image feature extraction and representation techniques. Int. J. Multimed. Ubiquitous Eng. (2013)
12. A. Ming, H. Ma, A blob detector in color images (2007), <https://doi.org/10.1145/1282280.1282335>
13. A. Chavan, Linear equation solver in Android using OCR. IOSR J. Eng. (2013). <https://doi.org/10.9790/3021-03514244>
14. A. Coates et al., Text detection and character recognition in scene images with unsupervised feature learning (2011), <https://doi.org/10.1109/ICDAR.2011.95>
15. Computer Algebra Systems, <http://www.math.wpi.edu/IQP/BVCalcHist/calc5.html>. Accessed 09 Sep. 2020
16. Joint Committee for Guides in Metrology (JCGM), Evaluation of measurement data—guide to the expression of uncertainty in measurement. Int. Organ. Stand. Geneva ISBN (2008), <https://doi.org/10.1373/clinchem.2003.030528>.
17. M. Mittal, R.K. Sharma, V.P. Singh, Modified single pass clustering with variable threshold approach. Int. J. Innov. Comput. Inf. Control **11**(1), 375–386 (2015)
18. M. Mittal, V.E. Balas, L.M. Goyal, R. Kumar (eds.), *Big Data Processing Using Spark in Cloud*, vol 43 (Springer Singapore, Singapore, 2019), <https://doi.org/10.1007/978-981-13-0550-4>

Usability Evaluation of Novel Text CAPTCHA Schemes Based on Colors and Shapes



Tejaswi Kumar, Navansh Goel, Siddhant Roy, and C. Oswald

Abstract Multiple web platforms are an imminent need of web security mechanisms to prevent vulnerable attacks, automated spammers and breach of data. One such security mechanism is to ensure that only authenticated users access them and one such cybersecurity algorithm is CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart). In this paper, we have designed novel and efficient text CAPTCHA schemes named MACS-TCHA (Manipulation of Alphanumeric characters, Colors and Shapes—Turing test to tell Computers and Human Apart). These CAPTCHAs are developed by manipulating alphanumeric characters, shapes and colors for better usability. Various user survey studies with different factors have evaluated our proposed CAPTCHA with significant improvement in terms of its usability.

Keywords CAPTCHA · MACS-TCHA · Usability · Evaluation · Security

1 Introduction

With the increasing use of web applications, there has also been an increase in the risks that these websites are facing. The increasing digitization has made security vulnerable. To increase security, many methods have been implemented in order to prevent giving access to unwanted users such as a bot. A bot, which is an abbreviation for a robot, is a kind of application software or a script that performs certain tasks.

T. Kumar · N. Goel · S. Roy · C. Oswald (✉)

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India
e-mail: oswald.c@vit.ac.in

T. Kumar

e-mail: tejaswi.kumar2019@vitstudent.ac.in

N. Goel

e-mail: navansh.goel2019@vitstudent.ac.in

S. Roy

e-mail: siddhant.roy2019@vitstudent.ac.in

Bots intended for malicious purposes allow an attacker to breach or take control of a site or a computer. To prevent these attacks, many security measures have been used to keep the sites and computers secured. One of the security measures used prominently around the world is the Turing Test, designed by a mathematician, Alan Turing. A Turing Test is a test for distinguishing between a human and a computer. In the presence of a judge, participants have to answer questions, based on which the judge decides between the two.

A CAPTCHA is a Completely Automated Public Turing test to tell Computers and Humans Apart. As the name suggests, these are automated programs that serve as the judge instead of a human [1]. CAPTCHAs pose a question to the user and by correctly answering the question; the system allows the user to proceed to the website. The questions are devised in a way so that only human users can identify the correct answer, keeping the bots out of the website [2]. There are various types of CAPTCHAs including visual, auditory and vocal senses of humans. These are Turing tests that ask the user to enter the answer in some form of text.

Nowadays, CAPTCHAs contain various loopholes, due to which each type of CAPTCHA has its own set of advantages and disadvantages. To reduce the errors of the existing CAPTCHAs and also to provide a better alternative, we have created a few versions of CAPTCHA named MACS-TCHA with shapes and colors. We have also evaluated their usability factors and substantiated that our MACS-TCHAs will not only provide security by protecting against bots but also will be easy for the users to solve.

The paper is organized as follows: Section 2 discusses the literature review of text and image CAPTCHAs. Our proposed MACS-TCHA schemes are explained in Section 3. In Section 4, we present the usability evaluation studies and discussions of our proposed schemes. Section 5 concludes with future directions.

2 Literature Survey

2.1 *Gimpy CAPTCHA*

As shown in Fig. 1, in this type of CAPTCHA, few words are selected and non-linearly modified. These are pasted on a distorted background and then presented to the user. Generally, the user is asked to type in any three words so that it becomes

Fig. 1 Example of Gimpy CAPTCHA



Fig. 2 Example of
NoCAPTCHA
ReCAPTCHA



difficult for a machine to solve the CAPTCHA [3]. This type of CAPTCHA can be easily solved by machines by using various OCR (**O**ptical **C**haracter **R**ecognition) techniques [4].

2.2 *NoCAPTCHA ReCAPTCHA*

Google's NoCAPTCHA ReCAPTCHA directly asks the user whether they are a robot or not and the user just has to click on the check box to pass the CAPTCHA as shown in Fig. 2. If the CAPTCHA is still unsure about the user, it checks and analyses the user's behavior of his previous, current and future actions [5]. This can be done using the user's browsing history as well as the movement of the user's mouse on the page. This CAPTCHA was concluded as the least frustrating CAPTCHA among the many available CAPTCHAs as per the survey conducted by Gafni et al. [6].

2.3 *Other CAPTCHAs*

There are various other CAPTCHAs like Text-to-Speech CAPTCHA [7], 3D CAPTCHA [7], Question-Based CAPTCHA [9] and DotCHA [10]. All these involve different types of representation of characters. On one hand, while 3D CAPTCHA displays characters with depth, the Question-Based CAPTCHA asks the user to enter some text based on the instruction provided. Other forms such as speech input or video input from the user can also be used to verify, as seen in Text-to-Speech CAPTCHA and Video CAPTCHA, respectively. The above-mentioned CAPTCHAs offer some sort of distortion of characters to the users. This could include overlapping of the characters, adding lines in the background, etc. These characteristics can prove to be time-consuming and difficult for users to solve [11]. To provide a better alternative to the existing CAPTCHAs, we have created a few CAPTCHA variations named MACS-TCHA, which are based on colors and shapes rather than focusing on the distortion of the characters.

3 Proposed Scheme

Our proposal is to create a CAPTCHA that is usable and can be solved by any authentic user with elementary knowledge. Recognizing entities such as shapes, colors, numbers, alphabets and performing basic mathematical operations are the underlying hypotheses of our work. This was achieved by using the randomness of these entities to develop our CAPTCHA named MACS-TCHA.

For each CAPTCHA, five character/digit images, five shapes, one instruction statement for the user, along with an input block and a submit button are displayed. Furthermore, on clicking the submit button, a message is displayed on the screen validating the user's input.

3.1 Algorithm

The following is the algorithm for the proposed MACS-TCHA.

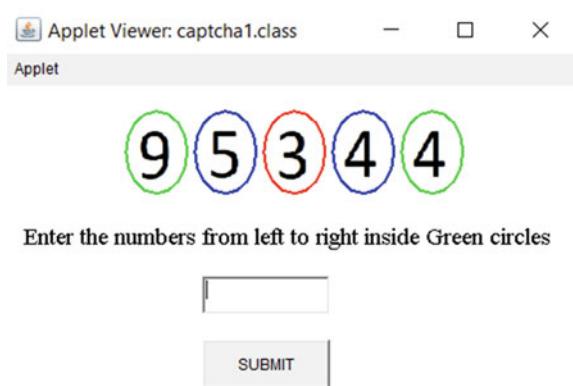
```
program Generator_MACS_TCHA()
n:= 5
Insert n random numbers into a static array ranging from 0 to 9
Insert numbers from 0 to 9 as key and their respective digit-image address as a
value into a hashmap
Create a hashmap for colors/shapes using their first letter as key and required
format as value.
Randomly select the instruction statement to display
According to the instruction set and static array, extract the correct answer.
Display the character images sequentially according to the static array and the
instruction statement
Take input from the user and validate it
If the user input is correct, display "Verified"
Else, display "Try Again", Reload the CAPTCHA and start the function from the
beginning
```

3.2 MACS-TCHA Scheme 1

In this scheme, a sequence of five random numbers is displayed to the user, as shown in Fig. 3. This contains a total of five numbers, each inside a colored circle. The color of these circles is randomly chosen from a total of three options, red, green and blue. The user will also be provided with an instruction statement. This instruction asks the user to enter all the numbers inside a certain colored circle.

Once the user enters the correct sequence, a "Verified" message is displayed and the user can be allowed to enter the website. If the sequence of numbers entered is

Fig. 3 MACS-TCHA Scheme 1

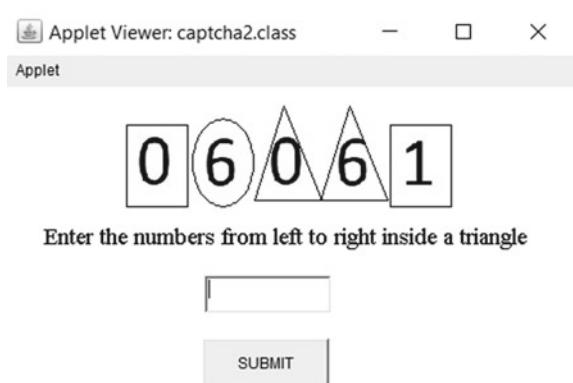


incorrect, a “Try Again” message is displayed along with which the entire sequence of numbers and the combination of colors get reloaded and an entirely new set is displayed to the user. Along with this, a new instruction statement will be provided and all the previously displayed data will be discarded.

3.3 MACS-TCHA Scheme 2

In this scheme, as shown in Fig. 4, a sequence of five random numbers inside shapes is displayed. The user has to enter the sequence of numbers sequentially, by following the instruction statement. For shapes, there are three options, which are triangles, squares and circles. The instruction statement will ask the user to type in numbers that are present inside a specific shape. After that, the user has to sequentially type only those numbers that are present inside the mentioned shape. The previously

Fig. 4 MACS-TCHA Scheme 2



displayed data will then be discarded. In case the user inputs the correct answer, a message will be displayed stating that the answer is correct.

4 Usability Evaluation Studies and Discussions

We have evaluated our MACS-TCHA using a comprehensive survey from various types of users, wherein a total of 204 responses were recorded. Out of these, 126 people were university students, 38 people were school students and 40 people were working professionals. Furthermore, the respondents were asked to choose the difficulty between easy, moderate and difficult, for two parameters, namely, Ease of Use and Clarity to Understand Instruction Statement.

4.1 Hit Ratio Versus CAPTCHA

The users were asked to solve the CAPTCHA, according to which the hit ratio was calculated using Formula (1). Figure 5 depicts the comparison between the hit ratio for both the CAPTCHA schemes.

$$\text{Hit Ratio} = \frac{\text{Total number of correct responses}}{\text{Total number of responses}} \quad (1)$$

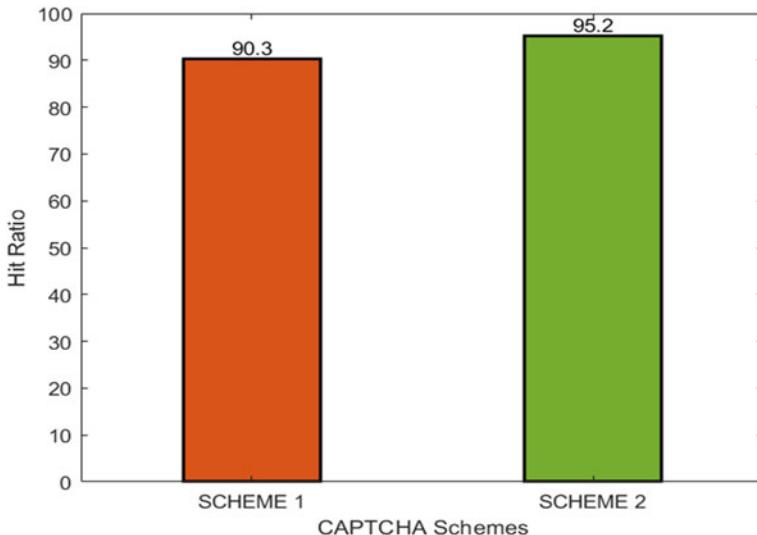


Fig. 5 Hit Ratio versus CAPTCHA Schemes

This helps us to understand the importance of shapes in contrast to colors, because shapes can be distinguished more accurately in comparison to colors, the ratio of people correctly answering the second scheme is higher.

4.2 Ease of Use Versus Profession for Both CAPTCHA Schemes

Most users found the CAPTCHAs easy in terms of ease of use. On plotting the graph, both schemes show positive results in terms of ease of use. As seen in Fig. 6, due to better recognition of shapes in comparison to color, scheme (2) showed slightly better ease of use.

The school-going users also found the CAPTCHA schemes easy or moderate. There were only a small number of users who found the CAPTCHA moderately difficult, thus aligning with our motivation to create this CAPTCHA.

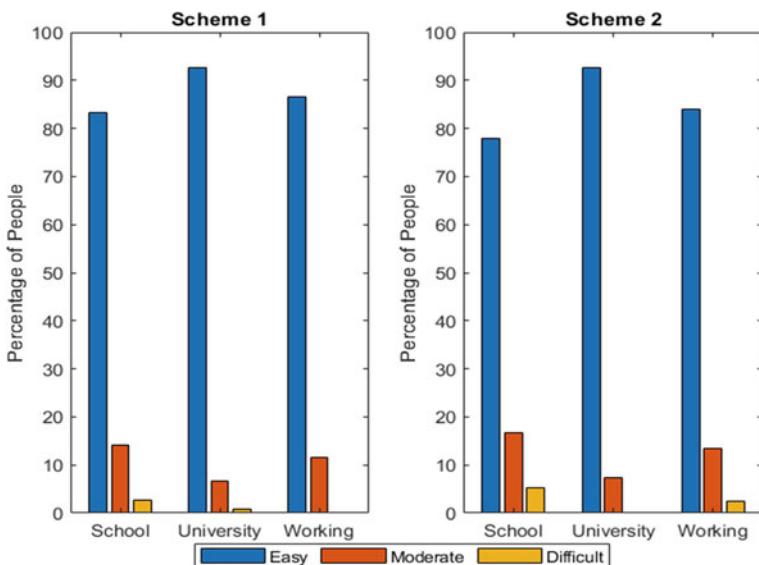


Fig. 6 Ease of Use versus Profession

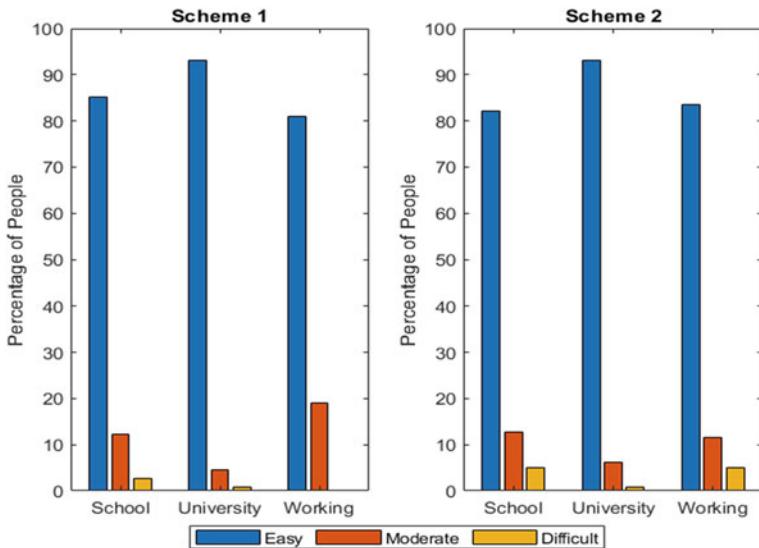


Fig. 7 Clarity in understanding Instruction Statement versus Profession

4.3 Clarity in Understanding Instruction Statement Versus Profession for Both CAPTCHA Schemes

Most users found the instruction statements of the CAPTCHAs easy to understand. On plotting the graphs, it is visible that in terms of understanding the instruction, all professions found the CAPTCHAs fit for real-time usage.

As seen in Fig. 7, both the working professionals and school-going users were able to understand the instruction statement clearly. The given instruction statement addresses the problem in a very basic manner that can be understood by all types of users.

5 Conclusions and Future Work

In this research, we have designed and developed various image CAPTCHAs by manipulating colors, shapes and alpha numerals so as to have better usability for the users. Our work evaluates the usability of our novel CAPTCHA named MACS-TCHA through a concise survey analysis. The survey studies have shown that the usability of the text-based CAPTCHA schemes has significantly improved after using the MACS-TCHA. As a part of ongoing work, we would like to design and develop more challenging variations of MACS-TCHA using multiple layers of randomization.

References

1. A. Dix, J. Finlay, G. Abowd, R. Beale, *Human Computer Interaction*, 3rd edn. (Prentice Hall, Hoboken, 2004)
2. L. von Ahn, M. Blum, J. Langford, Telling humans and computers apart (Automatically). *Commun. ACM*, **47**(2), 56–60 (2004)
3. K. Kaur, S. Behal, Captcha and its techniques: a review. *Int. J. Comp. Sci. Inform. Tech.* (2014)
4. G. Mori J. Malik, Recognizing objects in adversarial clutter: breaking a visual CAPTCHA, in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1 (2003), pp. 134–144
5. L. Ahn, B. Maurer, C. McMillen, D. Abraham, M. Blum, reCAPTCHA: Human-based character recognition via Web security measures. *Science (New York, N.Y.)* **321**, 1465–1468 (2008) <https://doi.org/10.1126/science.1160379>
6. R. Gafni, I. Nagar, CAPTCHA—Security affecting user experience. **905** (2016) <https://doi.org/10.28945/3469>
7. T.Y. Chan, Using a text-to-speech synthesizer to generate a reverse Turing test, in *Proceedings IEEE International Conference Tools with Artificial Intelligence (ICTAI 2003)*, pp. 226–232
8. M. Imsamai, S. Phimoltares, 3D CAPTCHA: a Next Generation of the CAPTCHA, in *International Conference on Information Science and Applications* (2010)
9. M. Shirali-Shahreza, S. Shirali-Shahreza, in *Question-based CAPTCHA Conference on Computational Intelligence and Multimedia Applications, International Conference on*, Sivakasi, TamilNadu, vol. 4 (2007), pp. 54–58
10. S. Kim, S. Choi, DotCHA: a 3D text-based scatter-type CAPTCHA, in *Web Engineering. ICWE 201*, Lecture Notes in Computer Science, ed. by M. Bakaev, F. Frasincar, I.Y. Ko, vol. 11496 (Springer, Cham, 2019)
11. E. Bursztein, M. Martin, J. Mitchell, Text-based CAPTCHA strengths and weaknesses, in *Proceedings of the 18th ACM conference on Computer and communications security* (2011), pp. 125–138
12. Y.W. Chow, W. Susilo, P. Thorncharoensri, CAPTCHA Design and Security Issues, in *Advances in Cyber Security: Principles, Techniques, and Applications*, ed. by K.C. Li, X. Chen, W. Susilo (Springer, Singapore, 2019)

Feature Selection for Email Phishing Detection Using Machine Learning



Neelam Yadav and Supriya P. Panda

Abstract Phishing attacks are one of the emerging and devastating cyber-attacks. Email is now considered as an appropriate means of written correspondence. Phishing emails are emails intended to extract sensitive and confidential information from the receiver. Two prong presentations are the aims of this paper; first, the analysis to remove the Phishing features from the email, and second, a notable triumph to reduce the complex data set to a lower dimension with different features. The feature reduction process is based on classification and prediction accuracy. The training and testing dataset is a collection of 1500 data tuples from the SPAMASSASIAN corpus. The validation dataset is made by retrieving the emails from Gmail users. To decide the two class codes, PHISHING and HAM, a total of 2000 emails are used to train, test, and validate the data tuples. In this research, 1000 emails are used to train, 500 emails are used to test the data tuples and validate them. First, the dataset is preprocessed to parse the data using HTML Parsing, Data Cleaning, Stemming, Stop Word Elimination, and Tokenization. By reading each email iteratively from the dataset, the features are extracted. Classifier ensemble strategies have gained the attention of many researchers in the machine learning research community in recent years [1]. Three machine learning classification algorithms are applied to predict the PHISHING and HAM emails such as decision trees (J48), random forest, and logistic regression. It was found that the random forest algorithm works best to separate PHISHING and HAM emails with the precision of a 99% classifier. With 15 feature sets, it fits best, the accuracy of training and validation is calculated to be 95.6% and 99.4%, respectively.

Keywords Zero-day Phishing · HAM · Classification · Stemming · Stop word removal · Tokenization · Supervised machine learning · Random forest · J48 and logistic regression · SPAMASSASIAN corpus · Dimension reduction · DOM · MIME

N. Yadav (✉) · S. P. Panda

Computer Science and Engineering, FET, Manav Rachna International Institute of Research and Studies, Sector-43, Aravalli Hills, Faridabad, Haryana, India

S. P. Panda

e-mail: supriya.fet@mriu.edu.in

1 Introduction

In today's time, the sender persuading in a phished email to provide personal information under pretenses is rampant. In such an attack, a user unknowingly reveals personal and confidential information to the invaders and then becomes the victim of the attack. The Phishing emails contain special links or notifications that encourage a user to respond to the email immediately by clicking on the links included in the email or giving the sender critical information. Phishing is imposing tough situations for layman users who do not know much about cyber-crime. Phishing is a criminal act of stealing sensitive information from a recipient for the benefit of an intruder. The most common example is that an attacker can send an email, which contains some infected hyperlink and this hyperlink takes one to a fake website that looks like a legitimate website such as snapdeal.com, citibank.com. Subsequently, the fake website asks for personal information like credit/debit card details, bank account numbers, user names, and passwords. By getting such information, they can attempt financial fraud. Phishing web pages are designed as same as legitimate web pages and then use for forging purposes.

Phishing features of an Email: Phishing is a continual threat. Phishing often takes place in email spoofing or instant messaging. Phishing email contains unrealistic demands or threats. It uses several phrases or words like "immediate action required", "your account is credited", "you won a phone", "lottery", "click the link for loan", which shows a sense of urgency to the user in order to take immediate action. One may ask to send the money for approvals, or loan sanctions, etc. after receiving such messages. Some of the main features of a Phishing email are the soaring count of the number of hyperlinks, and the number of images that serve as hyperlinks so these are the general features of Phishing emails. In this research, such features are extracted from the email dataset. These features are then trained using three machine learning classification algorithms to classify emails into Phishing and HAM.

In the remainder of this paper, Sect. 2 focuses on literature review addressing features to differentiate Phishing emails from HAM, Sect. 3 examines the proposed framework and architecture followed by preprocessing phase, feature extraction phase, and data preprocessing phase in Sects. 4 and 5, respectively. Subsequently, in Sect. 6, less important features are discarded using dimensionality reduction. Finally, in Sects. 7 and 8, classification models experiments, results, and analysis are stressed, respectively. The conclusion and potential scope are stated in Sects. 9 and 10, thus closing the paper.

2 Literature Review

The proposed studies use various features of Phishing attacks and aware the users, how to handle such Phishing attacks [2–6]. Researchers used several machine learning algorithms like SVM, random forest, and logistic regression to classify

HAM from Phishing mails. Form [1] focused on hybrid features of an email consist of URL-based, behavior-based, and content-based. The proposed method provides 97.25% accuracy and error rate of 2.75%. Akinyelu and Adewumi [3] worked on 15 features and used random forest classification algorithm. The method provided higher prediction accuracy (99.7%). Ranganayakulu [4] study was to detect malicious URL from an email by using lexical and host-based features. An email server named SSE Mail Server is used for testing purposes. The dataset is obtained from DMOZ open directory and Phistank.

Vazhayil et al. [7] used classical machine learning techniques to the data in order to classify an email as Phishing or legitimate. Khonji et al. [8] used 43 features to perform precision and recall analysis. They use six classifiers from which random forest was considered the best after CART. A total of 2889 emails are used wherein 1171 are Phishing and the rests are HAMS.

Almomani et al. [9] study focused on the “Zero-Day” Phishing emails. They worked on the detection and prediction of “Zero-Day” Phishing emails by using the Evolving Fuzzy Neural Network (EFuNN). The dataset has 2000 HAM emails from the monkey website and 2000 from SPAMASSASSIN. Smadi et al. [10] also focused on fundamental attacker behavior. This behavioral information is extracted from email header. This approach is based on a combination of content and behavior. These hybrid feature selections are able to achieve 96% accuracy rate. Patel and Mehta [11] proposed that hybrid features should be used, which should include behavioral and content-based to detect fraud emails. Rathod and Pattewar [12] proposes a content-based email classification model using Naïve Bayes and C4.5 decision tree. The integrated approach provides 95.54% accuracy. Kumar et al. [13] proposes a DC scanner. This is an email scanner that identifies the malicious URL from an email message. Form et al. [5] propose to use hybrid features to differentiate Phishing emails from HAM. These hybrid features contain content-based, URL-based, and behavior-based features. In total, 1000 emails are used to train and test the model. The author focuses on nine features and uses SVM to classify. The model achieved 97.25% accuracy.

Pandey and Ravi [14] propose an intelligent preprocessing phase that extracts different email parts. A total of 23 features are identified and the J48 decision tree algorithm is used in classification. Tenfold cross-validation is used. The model attained 98.87% accuracy in the random forest model. Ozarkar and Patwardhan [15] use seven feature selection methods, for instance, Chi-square, Information Gain, Gain Ratio, Relief, Symmetrical Uncertainty, OneR, and Correlation. Support Vector Machine classification model is used and achieved above 97.7% accuracy using all the seven feature selection methods. Nine features are used for feature selection. Anh et al. [16] propose a study based on fuzzy for Phishing website detection. The authors have used 11,660 sites for training purposes. The fuzzy techniques gave 99.25% accuracy. Vaithianathan et al. [17] have used J48, Naïve Bayes, Bayes Net, and Multilayer Perceptron classification techniques and give a comparison of three different datasets (weather, labor, and soybean). J48 and Naive Bays classifier’s accuracy is efficient. Youn and McLeod [18] propose the use of ontology. Ontologies help to understand the semantics of data. The classifier’s accuracy is 97%.

3 Proposed System and Architecture

3.1 Phases in System Design

Figure 1 depicts the proposed architecture to extract the Phishing terms, features, which helps to make an intelligent classifier to detect the Phishing emails. To identify the Phishing terms, a raw email message is first preprocessed to remove the unnecessary contents (like stop words removal, stemming, to retrieve the inner text of HTML tags, etc.) and to smoothen the data.

The email contents are mined to discover information/knowledge so that a classifier can be trained, which helps in identifying the Phishing email. To identify the ant-Phishing features, the concept of term-document frequency is applied to the contents of a new email message. This knowledge mining process will be carried out using the following steps.

1. The email messages are first converted into “.eml” extension. The email messages are then preprocessed in the data preprocessing phase shown in Fig. 2 in order to remove the outliers or inconsistencies present in the data.
2. Next, a Java program is written to retrieve the email headers, like Multipurpose Internet Mail Extension (MIME) Type, Subject, Date, from, to and Body of the email, etc.
3. The most frequent word is identified from each email and from these words the first feature set with features was made. These initial features(306) were

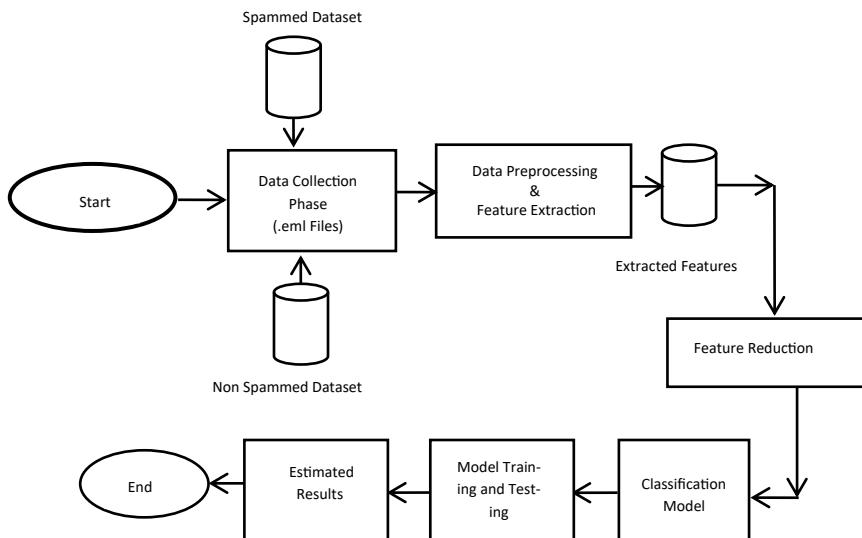


Fig. 1 The proposed model architecture of anti-Phishing feature selection

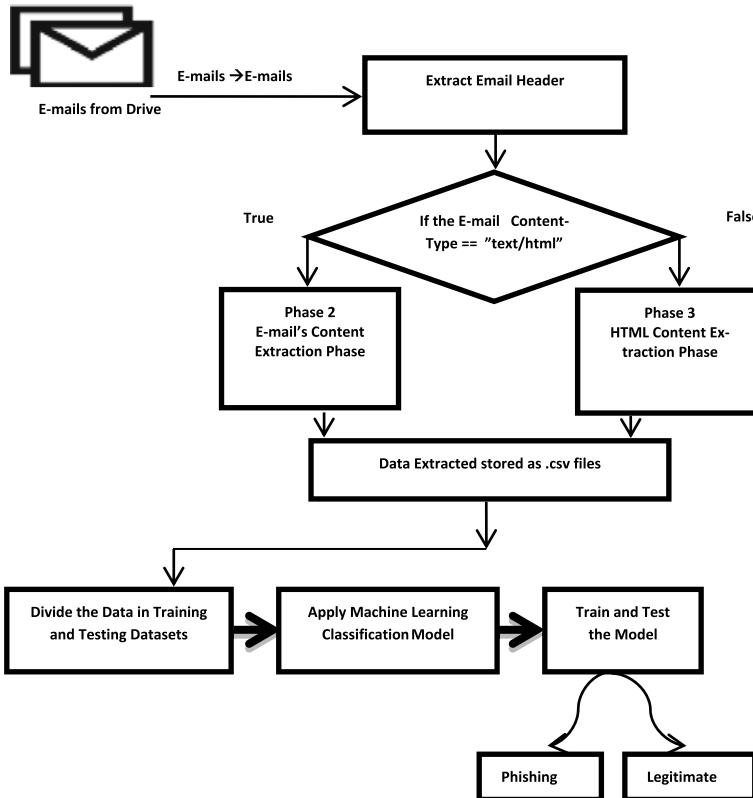


Fig. 2 System model for anti-Phishing feature selection

- obtained by calculating the total number of occurrences of specific words in an email. The most frequently occurred word is made as a feature of an email.
4. The weight of Phishing terms (features) is calculated using termed frequency (TF) that helps in discriminating a Phishing email from legitimate emails.

3.2 Feature Reduction

The proposed system model in Fig. 2 reduces the feature dimension to 50, then 20, and finally to 15 only, which helps to improve the classification accuracy. The dimensions of features are reduced as it helps in removing the noise, which occurred while training the classifier. The number of features is reduced by calculating the term frequency-inverse document frequency (TF-IDF) of the feature in the dataset. Information gain is also used to reduce the feature set.

After reducing the dimension, the classification process is done in WEKA. Different classifiers are used here like logistic regression (LR), J48 and random

forest. The classifier is first trained by providing the training data and then the classification model is tested and validated. Several metrics are used to evaluate the classification model.

3.3 Data Collection Phase

To build an email Phishing classifier, a collection of sample emails are taken. The sample emails are a collection of legitimate (HAM) and Phishing (spam) emails.

Data Cleaning The dataset is first cleaned by removing the noise and outliers from the dataset, and only legitimate and HAM emails are considered.

Data Integration The dataset is integrated from multiple data sources, which are combined, for both Phishing as well as HAM emails. The Phishing 4559 emails were collected from Nazario [6] and 4559 were selected from the SPAM ASSASSIN project [19]. Out of these emails, 1500 emails are collectively used to train the classifiers.

Data Selection Our training and testing dataset entails 1500 mails. This dataset pool includes 750 HAM emails from the SPAM ASSASSIN project [6, 19] and 750 spam emails from CSDMC2010 SPAM corpus. A Java program is written to smooth the emails. Emails are then converted to “.eml” extension and stored separately in different folders, SPAM folder for spammed emails and the HAM folder for non-spammed emails.

4 Data Preprocessing Phase

The Data Preprocessing: The data preprocessing phase in Fig. 3 is carried out using JAVA. It is a data cleaning phase. In this phase, the email headers like Subject and Body are extracted for each training dataset email. This phase first parses the data using HTML text parsing. Then the data cleaning phase is done to remove the extra symbols. The data tokenization breaks the content of the Subject and Body into words. Because the Phishing terms are identified using TF-IDF measure in which the frequency of occurrence of the word is identified. To do so, intelligent and important steps are used to stop word removal and stemming. The data preprocessing phase comprises the following steps:

- i HTML parsing: The body and subject headers from the email message are extracted. If the email’s content type is text/HTML then HTML parsing is necessary. HTML parsing is a process of extracting data from the HTML form. The HTML document is treated as Document Object Model (DOM) objects. JSOUP technology is used to parse the HTML content. Here JSOUP is used to retrieve the inner text of the HTML tags. Many email content features are

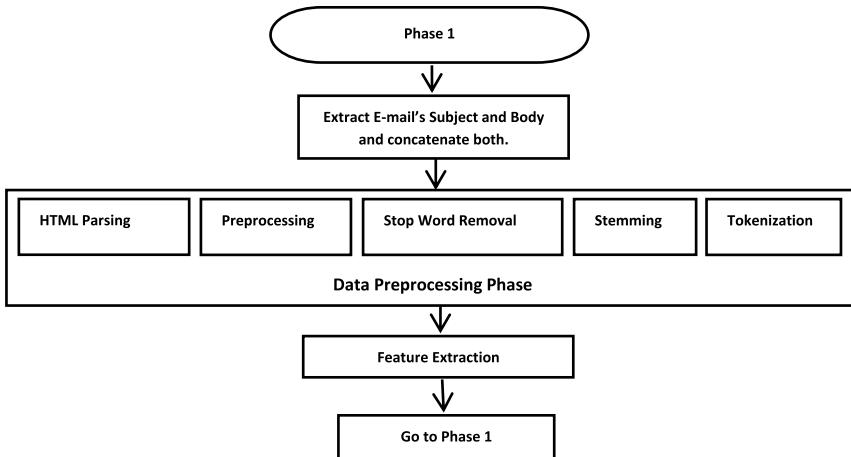


Fig. 3 Data preprocessing Phase 1

computed by using HTML Parsing. This can be achieved by fetching the HTML elements and their inner contents. After HTML parsing, the data are cleaned and the HTML tags are removed from the email content.

- ii Text Preprocessing: The Emails' message body and subject are concatenated before preprocessing. These data are then processed and cleaned by removing the line breaks, extra symbols like (>, \n, </, , <, \t, |, ♦, ♦ =, _, -, /, #, ==, &, •, :, @, !) etc. These special characters need to be removed so that the data can be cleaned and tokenized properly.
- iii Stop Word Removal: In this step, the stop words are removed from the parse and preprocessed email message. Stop words are helping verbs, pronouns, articles, and prepositions, which are used to make a sentence and these words are mostly removed.
- iv Stemming: It is the process of reducing a word to its stem or root form. Using stemming, one may find search results for any word that contains the root form of the word.
- v Tokenization: After processing the subject and body text, these are then extracted as tokens (words) by splitting the email body (line of text) with each occurrence of space (" "); Tokenization also enhances the stop word removal and stemming process.

5 Feature Extraction Phase

The feature extraction phase is divided into two parts.

- (a) Email's HTML content extraction phase
- (b) Extraction of the phishing terms from the email body and subject.

The email has two basic parts, the email header and the body. The header contains information such as the message sender, receiver, message-ID, MIME-Version, date, and content type. The message content contains substantive information intended for those who read the message. The features are extracted by counting the individual words in an email. In each email, the maximum occurrence of a word is identified. So in total, 3727 words are identified and then these words are matched with the words that the researchers have already studied. From these, we use term frequency-inverse document frequency (TF-IDF) to select 307 features to be used for further analysis. These features are collected from previous studies [2–5].

5.1 Email's HTML Content Extraction Phase

HTML text parsing: The email's header, body, subject, and inner text are extracted from the training dataset emails one by one using the script in Java, and JSOUP technology is used to parse the HTML file. The HTML parsing step in Fig. 4 helps identify the first five features, which are extracted from an email's content and behavior. The first five features are as follows:

- (a) “*isHTML*” An email's body is checked to see whether or not this content is HTML. “*isHTML*” is stored as {0, 1} in the features dataset. If the email has the content type as “text/html” then it is stored as 1 otherwise an email is stored as 0 in the features dataset.

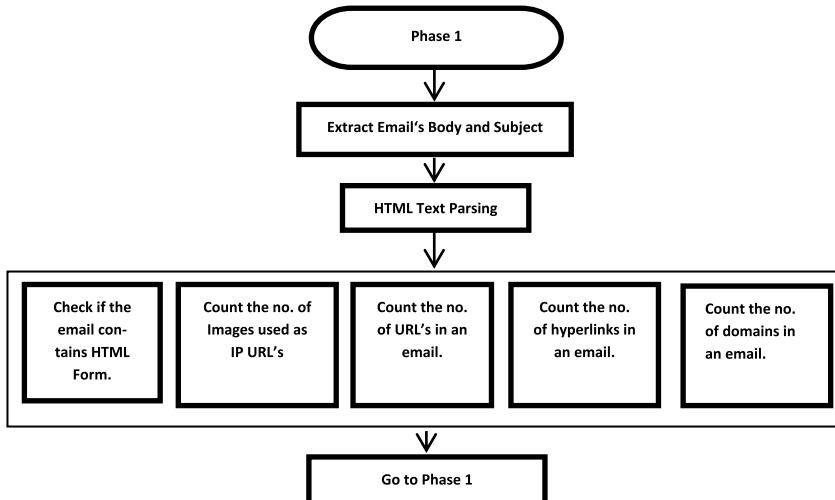


Fig. 4 HTML content extraction phase: Phase 2

- (b) “*NoofURLs*” It is a content feature of an email. It is the count of the number of URLs or IP addresses in an email message. “*NoofURLs*” is a numeric feature. It is stored as a count of URL’s in an email in the features.
- (c) “*NoofDomains*” This feature counts the number of different domains, which is used as hyperlinks in the email. “*NoofDomains*” is computed as the count of number of unique domains in the features dataset.
- (d) “*NoofHyperlinks*” It is the count of number of hyperlinks in an email body. “*NoofHyperlinks*” is computed as the count of hyperlinks in the features dataset.
- (e) “*ImagesAsURLs*” It is a content feature that is computed as the number of images in an email message that is used as hyperlinks. “*ImagesAsURLs*” is computed as number of images used as hyperlinks in the features dataset.

5.2 *Phishing Terms Extraction from the Email Body and Subject*

TF-IDF is used to determine frequent terms in an email whose frequency is very high; it is a measure of numerical statistics to determine the significance of a word in a document or in an email. The Phishing terms in Table 1 are those that have the highest TF in the Phishing dataset. “tf-idf” describes the value of a term. The TF-IDF weight is used to discover interesting patterns in text mining and to retrieve important information. Phishing terms are those key terms that exist in certain Phishing emails for instance, the word **click, credit, grants, offers, free, insurance and financial, etc.** [2–4, 20].

6 Dimensionality Reduction

Dimensionality Reduction The occurrence count of each feature inside an email is used as 307 feature attributes of that email. Less important features are discarded. The 50 features are selected by calculating the TF-IDF in order to check the efficiency of the feature selection methods. The final 15 features in Table 1 are selected out of the 50 features using the dimensionality reduction algorithm. The logistic regression classifier is performed using the selected features over the same email set.

Dimension (Feature) Reduction Algorithm

Input: Distinct frequently occurred features from the email

Algorithm:

1. Start with the key features.
2. Apply TF-IDF on the most frequent key features.

Table 1 Feature set with 15 features

ImagesAsURLs	Price
NoofURLs	Business
NoofDomains	Credit
NoofHyperlinks	Offers
isHTML	Money
Grants	Wish
Click	Insurance
Reply	

3. Features with lower weights are ignored and the features with higher weights are retained.
4. Create a classification model from these features.
5. Do the testing and validation with the classification model.
6. Check the model's testing and validation accuracy.
7. If the results are satisfying then terminate the loop.
8. Else
9. Repeat the steps from [2 to 7].

Output: Reduced feature set with high accuracy.

7 Classification Models

Classification algorithms can be used in the way they learn about the patterns and data to make predictions. Classification models include logistic regression, decision tree, random forest, multilayer perceptron, and Naive Bayes. Logistic regression [21] is a binary classification algorithm, which works on categorical class labels where the variables are binary that is where the output variable should have two possible values like (HAM or PHISHING, Yes or No, buy or not buy).

8 Experiments, Results, and Analyses

For the given two classes, the **positive tuples** (e.g., HAM) versus **negative tuples** (e.g., Phishing), a confusion matrix for positive and negative tuples is shown in Fig. 5. This figure is a screenshot of research work in WEKA. In this research, two class labels are identified as HAM and Phishing. The training dataset has 1000 records. This training set consists of HAM and Phishing-featured emails as 500 and 500, respectively.

==== Confusion Matrix ===		
a	b	<-- classified as
497	3	a = HAM
10	490	b = PHISHING

Fig. 5 Confusion matrix for HAM and PHISHING emails

Dataset Preparation and Result Analysis: Here, the widely used SpamAssassin corpus's email database [19] is used in experiments. This database contains a wide variety of spam emails that guarantees the proposed feature selection methods to be suitable for different types of emails. The number of emails in the training, test sets, and validation datasets is defined as 1000, 500, and 500, respectively for both spam and non-spam email classes. A total of 15 features are experimented by J48, random forest, and logistic regression classifiers. The training, testing, and validation accuracy are described in Table 2. In total, 500 emails are used as validation. The results of the validation are described in Table 2. To create a validation dataset, the spam emails from Gmail are collected from 10 different users. These mails are downloaded manually without using any tool. The HAM emails are retrieved from a college email.

Table 2 shows the classification results for all three classification techniques. The data tuples are tested and validated on the training models and the performance comparisons of J48, random forest (RF), and logistic regression (LR) using 15 features are shown in Fig. 6.

9 Conclusions

The number of unwanted emails called spam has created an urgent need for more accurate and robust antispam growth [22]. Phishing functions assist in the identification of Phishing emails and also help to determine an anti-Phishing classification system. The technique approached with Phishing and HAM emails distinguishes the new emails from the Phishing emails. A smart and intelligent data preprocessing technique is used to sanitize the email data. The preprocessing includes HTML Parsing, Stop Word Removal, Stemming, and Tokenization. The teething troubles with PHISHING emails are a well-known fact. The training, testing, and validation are done on WEKA. The features are extracted by reading each email iteratively from the dataset.

Initially, 307 Phishing words are extracted by calculating the most frequently occurred words in the emails. As the testing accuracy was not up to the mark, hence feature reduction techniques were used to reduce the feature set. To reduce the features, the TF-IDF is used iteratively and the features with higher values are taken

Table 2 J48, random forest, and logistic regression classification and prediction result
Classification applied through J48, random forest, and logistic regression

				J48			Random forest			Logistic regression		
Tuples in training dataset	Tuples in testing dataset	Tuples in validating dataset	Number of features	Training Accuracy	Testing Accuracy	Validation Accuracy	Training Accuracy	Testing Accuracy	Validation Accuracy	Training Accuracy	Testing Accuracy	Validation Accuracy
1000	500	500	15	96.9	96.2	88	98.7	95.6	99.4	95.5	98.2	97.8



Fig. 6 Performance comparison of J48, random forest, and LR using 15 features

and rests are discarded. Finally, three varied feature sets are identified for instance 50, 20, and 15 on which three machine learning classification algorithms such as decision trees (J48), random forest, and logistic regression are applied to predict the PHISHING and HAM emails. It was observed that the random forest algorithm works best to distinguish the PHISHING and HAM emails with a 99% classifier's accuracy. It works best with 15 feature sets. The training and validation accuracy is determined as 95.6% and 99.4%, respectively.

10 Future Scope

The number of unwanted emails called spam has created an urgent need for more accurate and robust antispam growth. To upsurge the effectiveness of random forest algorithm in terms of testing accuracy, there is a necessity to extract more features from an email, for instance, “number of the receiver”, “text link difference”, “domain name difference in sender’s and inside the email’s body”, “use of <script> tag”, etc. These added features can be used to make the PHISHING email detection more fruitful. In future work, new Phishing terms can be identified with high TF-IDF values to reinforce the likeness between Phishing email terms. The proposed model can further be boosted by adding the new features and by testing the current feature sets with other classification models for instance Naïve Bayes, Support Vector Machine(SVM), to name a few. Also, for reducing the dimension, a Modified Whale Optimization Algorithm (MWOA) for software usability feature extraction [22] can be used.

References

1. O.A. Alzubi, J.A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, M. Ramachandran, An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* (2020)
2. L.M. Form, K.L. Chiew, S.N. Sze, W.K. Tiong, Phishing email detection technique by using hybrid features, in *Proceedings of the 9th International Conference on IT Asia (CITA)*, Aug. 2015, pp. 1–5
3. A.A. Akinyelu, A.O. Adewumi, Classification of Phishing email using RF machine learning technique. *J. Appl. Math.* (Hindawi Publishing Corporation) **2014** (2014), Article ID 425731
4. D. Ranganayakulu, C. Chellapan, Detecting malicious URLs in E-mail—an implementation, in *Proceedia of ASRI Conference on Intelligent Systems and Control* (Elsevier, 2013), pp. 125–131
5. L.M. Form, K.L. Chiew, S.N. Sze, W.K. Tiong, Phishing email detection technique by using hybrid features, in *9th International Conference on IT in Asia (CITA)* (2015)
6. J. Nazario, Phishing Corpus, <https://monkey.org/~jose/Phishing/>. Accessed June 2016
7. A. Vazhayil, N.B. Harikrishnan, R. Vinayakumar, K.P. Soman, PED-ML: Phishing email detection using classical machine learning techniques, in *Proceedings of the 1st Anti-Phishing Shared Pilot 4th ACM International Workshop Security and Privacy Analytics (IWSPA)*, ed. by A.D.R. Verma (Tempe, AZ, USA, 2018), pp. 1–8
8. M. Khonji, Y. Iraqi, A. Jones, Enhancing Phishing e-mail classifiers: a lexical url analysis approach. *Int. J. Inf. Secur. Res. (IJISR)* **2** (2012)
9. T.-C. Almomani, A.A. Wan, et al., Evolving fuzzy neural network for Phishing e-mails detection. *J. Comput. Sci.* **8**(7), 1099–1107 (2012)
10. S. Smadi, N. Aslam, L. Zhang, R. Alasem, M. Hossain, Detection of Phishing e-mails using data mining algorithms, in *9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)* (IEEE, 2015), pp. 1–8
11. J. Patel, S. Mehta, A literature review on Phishing email detection using data mining. *Int. J. Eng. Sci. Res. Technol.* **4**(3), 46–53 (2015)
12. S.B. Rathod, T.M. Pattewar, A comparative performance evaluation of content based spam and malicious url detection in e-mail, in *IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)* (2015)
13. B. Kumar, P. Kumar, A. Mundra, S. Kabra, DC scanner: detecting Phishing attack, in *IEEE Third International Conference on Image Information Processing* (2015)
14. M. Pandey, V. Ravi, Detecting Phishing e-mails using text and data mining, in *IEEE International Conference on Computational Intelligence and Computing Research* (2012)
15. P. Ozarkar, M. Patwardhan, Efficient spam classification by appropriate feature selection. *Int. J. Comput. Eng. Technol. (IJCET)* **4**(3), May–June (2013). ISSN 0976-6375 (Online)
16. L. Anh, T. Nguyen, H.K. Nguyen, Developing an efficient fuzzy model for Phishing identification, in *10th Asian Control Conference (ASCC)* (IEEE, 2015), pp. 1–6
17. V. Vaithianathan, K. Rajeswari, K. Tajane, R. Pitale, Comparison of different classification techniques using different datasets. *Int. J. Adv. Eng. Technol.* **6**(2) (2013)
18. S. Youn, D. McLeod, Efficient spam email filtering using adaptive ontology, in *Fourth International Conference on Information Technology, ITNG'07* (IEEE, 2007), pp. 249–254
19. “SPAMASSASSIN data”, Spam-Assassin datasets, Csmining Group, <http://www.csmining.org/index.php/spam-assassin-datasets.html>. Accessed 23 February 2018
20. A. Yasin, A. Abuhasan, An intelligent classification model for Phishing email detection. *Int. J. Netw. Secur. Appl.* **8**(4), 55–72 (2016)
21. J. Brownlee, Logistic regression tutorial for machine learning. Machine Learning Mastery, 06 November 2016, <https://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/>. Accessed 26 May 2018
22. R. Jain, D. Gupta, A. Khanna, Usability feature optimization using MWOA, in *International Conference on Innovative Computing and Communications (ICICC)*, ed. by S. Bhattacharyya, A. Hassani, D. Gupta, A. Khanna, I. Pan (2018)

Workflow Scheduling Using Optimization Algorithm in Fog Computing



Gaurav Goel, Rajeev Tiwari, Abhineet Anand, and Sumit Kumar

Abstract In the cloud computing paradigm data, owners have to put up their data in the cloud. Due to the longest distance between devices and cloud; problem of delay, bandwidth, and jitter is there. Fog computing was introduced to the edge of the network to overcome cloud problems. During the transfer of data between the Internet of Things (IoT) devices and fog node, scheduling of resources and tasks is necessary to enrich quality of service (QoS) parameters. Various optimization and scheduling algorithms were implemented in a fog environment. Still, the fog environment is facing the problem of efficiency, latency, cost, computation time, and total execution time. Earlier PSO (particle swarm optimization) techniques or ACO (ant colony optimization) are provided the solution to NP-hard problems. Over such types of optimization techniques, various optimization algorithms are provided like Dolphin Partner optimization, Grey wolf, Moth-Flame, Firefly, crow, etc. Priority queue, round robin scheduling algorithm implemented on another side for a solution to the problem. In this paper, the implementation comparison of PSO, ACO on the cloud, and Fog is contrasting using iFogSim toolkit. The results of QoS parameters makespan and cost in fog computing are showing enhancement in QoS over cloud computing.

G. Goel
UPES Dehradun, Dehradun, India

CGC Landran, Landran, India

G. Goel
e-mail: gaurav.coecse@cgc.edu.in

R. Tiwari (✉) · S. Kumar
School of Computer Science, UPES Dehradun, Dehradun, India
e-mail: rajeev.tiwari@ddn.upes.ac.in

S. Kumar
e-mail: sumit.kumar@ddn.upes.ac.in

A. Anand
CUIET–CSE, Chitkara University, Chandigarh, India

Keywords Resource scheduling · Optimization algorithm · Fog computing · QoS · IoT

1 Introduction

In the cloud computing paradigm, data owners have to put up their data from the local side to the cloud server side. Cloud computing is used in many applications such as social networks, defense, medical science, road network, etc. Cloud computing is providing on-demand resources, storage, and computing power. Cloud computing typically exists in a backend data center, with data being scattered from more or less centralized resources (e.g. compute, storage) to consumers at the network edge. Fog provides computation, decision-building, and action-taking to happen via the Internet-of-Things devices and only post applicable data to the cloud. As shown in Fig. 1, fog nodes can be distributed anywhere with a network connection: on an industry, ground, on top of a power pillar, by the side of a railway track, in traffic, or on an oil rig [1–4]. Any appliance with a network connection, computation, and repository can be a fog node. Fog is acting as a mediator to bring the Internet-of-Things to human life by providing goods and computer functions and creating a middle layer that enables in between IoT devices and cloud devices. The purpose of scheduling the resources is to discover better resources for customers to realize excellent planning goals, such as lower processing delay and enhance resource utilization and quality-of-services (QoS). For the sake of solving various NP-hard problems of resource scheduling/task scheduling, in the traditional systems, many optimization algorithms have been proposed [5–9].

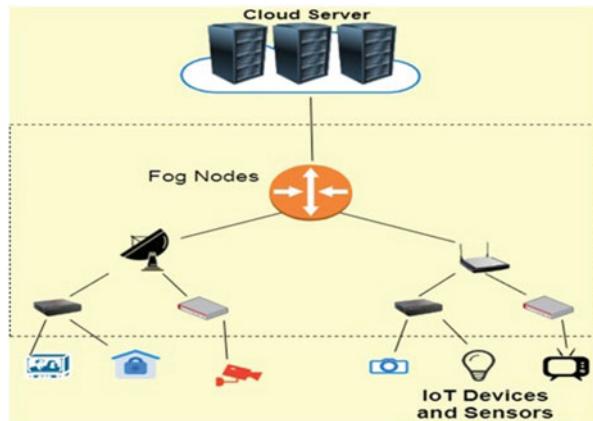


Fig. 1 Fog computing environment

1.1 Challenges

- As in comparison with the cloud environment, fog is providing better bandwidth and less delay in the system, but still, segregation of jobs into tasks is a great headache.
- The assignment of tasks to a particular fog node is another challenge.
- Resource scheduling in a system is still a big challenge.

1.2 Motivation

- For the improvement of QoS, many scheduling and optimization techniques are introduced; still, many algorithms are showing delay in data transmission, latency problem, and bandwidth, etc.
- Resource scheduling is a technique where; by doing task scheduling and resource allocation, these problems of QoS parameters can be overcome.
- The motivation of designing an effective optimization algorithm for resource scheduling will improve QoS in the fog environment.

1.3 Organization of the Paper

Section 2 shows significant work done in this area, which shows proposed proposals. In Sect. 3, an algorithm is written for PSO and ACO. In Sect. 4, results are evaluated. Section 5 is about the conclusion and future scope.

2 Related Works

Sun et al. [1] suggested the latest multifaceted optimization problem known as an enhanced NSGA-2 algorithm that effectively reduces service efficiency and improves the overall efficiency of project scheduling, which is divided into extremely layer, cross-layer, and central layer. Bitam et al. [2] suggested the latest bio-inspired optimization approach known as bees life algorithm (BLA), directing in the fog environment is the problem of job scheduling. The system consists of the administrative node, which is responsible for job scheduling after each job is received from mobile users. After the allocation of the job, each fog node will execute the allocated task. Author implements bees life algorithm for job scheduling to optimize the problem of CPU execution time and allocated memory for the task. Ghobaei-Arani et al. [3] proposed algorithm likewise a particle swarm optimization; initialize a population of moth randomly and compute objective function, second creating a set of flames same as moth solution and update the position of moths and change the flame size. Hussien

and Mousa [4] proposed a task offloading mechanism in fog computing using an ant colony optimization algorithm, two algorithms particle swarm optimization and ant colony optimization have been proposed for job scheduling for improving response time and communication cost in a system. Both algorithms were then compared with the round robin algorithm and improvement shown for response time and communication cost over the RR algorithm. Senthil Kumar and Kasireddi [5] proposed a combination of the firefly and crow algorithm and shown maximize makespan and maximize throughput of a system over individual firefly and crow algorithm. The advantage of the system is two algorithms combined, which helps improve the global search capability and completion time of a system. Li et al. [6] proposed the methods of resource scheduling based on optimized fuzzy clustering in which a resource required is clustered into the basis of three parameters, computation, bandwidth, and storage, and then provide required resources to desired sensor data from cluster after matching resources using FCAP algorithm. Particle swarm optimization algorithm is used along with fuzzy clustering technique. Rehman et al. [7] proposed a resource distribution using the Min-Min algorithm for cloud and fog in smart buildings. The author also proposed concepts of smart grid; through which calculation of energy is required for six clusters having 20 buildings calculated and desired VM assigned to cluster for performing the task. As the Min-Min algorithm is used so task having low computation time getting VM first and a task having the longest computation time placed in the queue. Nazir and Shafiq [8] suggested scheduling of jobs using fog and cloud computing in smart grid by using the Cuckoo optimization algorithm. The author works on a technique to balance resource management, reaction time, and vulcanization time and with the help of a cuckoo optimization algorithm. Choudhari et al. [9] proposed a prioritized task scheduling in fog computing. Each incoming request is assigned to the nearest Fog Server and the request will be placed in the priority queue for completion, if the nearest FS not available then it is forward to another nearest Fog Server and if resources not at all available then the request will forward to cloud server. Yin et al. [10] suggested scheduling of tasks and allocation of resources in fog based on container approach for on-time completion of tasks and the number of tasks for the fog node is optimized. Authors used the technique of accepting and rejecting tasks, if tasks accepted then it forward to either fog or cloud depend upon several resources required by the task. Unlike VM, containers used in a system and modifying resource quota of containers so that resource quota of tasks can be changed called resource reallocation and it does not create any delay overhead. Gao et al. [11] proposed an energy-efficient scheduling algorithm on heterogeneous distributed systems. Author proposed a DVFS and turning off the appropriate processors to reduce dynamic and static energy consumption. Ambe et al. [12] provided QoS aspects of the economy for fog computing having reliability, performance, and cost. Reliability has further service continuity and network quality. Achieving continuous service delivery and migration plays a key role. Wang and Batiha [13] proposed a comprehensive study on methodology in the fog environment. In this article, the author writes management policies in three main categories: data, energy, and resource management. Mohammed et al. [14] proposed a green energy source: issues and challenges in this paper. Mobile offloading is a concept based on the idea

of exchanging relatively low communication energy for high computation power utilization. Haghi Kashani et al. [15] proposed a taxonomy for QoS aware techniques and contrast comparison on application management, service/resource management, and communication management for 11 QoS parameters, cost, deadline, availability, energy consumption, execution time, reliability, throughput, response time, resource utilization, scalability, and security. Pereira et al. [16] proposed a scheme for fog computing applications and services in a VANET's environment. Author works on two types of fog applications, one to detect traffic anomalies such as traffic, road works, and other mobility issues and the second to estimate bus time to feed traveler information. Toor et al. [17] proposed a technique of dynamic voltage frequency scaling and green renewable energy. Energy consumption for fog nodes is a great headache in today's era. Green renewable energy is a way to charge fog nodes. The author proposed scheme is to check weather conditions, whether to charge nodes with renewable energy or with non-renewable energy. Tiwari et al. [18] suggested the idea of compiling web cache, with good standardization, with statistical dimensions of a first-party representative, is a representative force when congestion is exceeded. This database verified similar information between the proxy cache objects and original server objects. Sharma et al. [19] proposed big data application analytics. Authors made a comparison between real-time analytics, size, and database used using open source tools such as kibana, elastic search, and Jason query. Luo et al. [20] proposed an algorithm that guarantees energy balancing of terminal devices without enhancement of transmission delay. Lal et al. [21] proposed an allocation and scheduling policy between virtual machines in the cloud environment using cloudsim and shown performance on parameters makespan, execution, and throughput of allocated tasks. Tiwari and Kumar [22] proposed a web caching algorithm, works on parameter hit ratio and latency time, and shows an improvement of the proposed scheme with the existing scheme. Peralta et al. [23] proposed a scheme to reduce downloading time for the end device. In this context after fusing, fog and cloud computing mean a suitable solution. The author proposed an optimal distribution algorithm that regulates the quantity of data to be stored or retrieved to reduce data placement time (Table 1).

3 Particle Swarm Optimization and Ant Colony Optimization Algorithms

Algorithm Particle Swarm Optimization

1. **Loading:** Assign new position and velocity value to each particle.
2. **Exchange the position of each particle:** Exchange the rapid location vector to a discrete vector.
3. **Fitness Value:** The value of fitness of each particle will be computed from the fitness function.

Table 1 Comparison table of previous algorithms

Approach	Year	Technique used	Advantage
Bitam et al. [2]	2018	Bees swarm optimization	The advantage of the system is that the author implements bees life algorithm for job scheduling to optimize the problem of allocation of memory for tasks and CPU execution time
Ghobaei-Arani et al. [3]	2020	Moth Flame Optimization	The advantage of the system is likewise a particle swarm optimization; initialize a population of moth randomly and compute objective function, second creating a set of flames same as moth solution and update the position of moths and change the flame size
Hussien and Mousa [4]	2020	Ant colony optimization	The advantage of the system is an adaptive method provided by the author for the calculation of response time and communication cost for particle swarm optimization and ant colony optimization algorithms
Senthil Kumar and Kasireddi [5]	2019	Firefly and crow algorithm	The advantage of the system is two algorithms combined, which helps improve the global search capability and completion time of a system
Li et al. [6]	2019	Fuzzy clustering: cluster of storage, computation, and bandwidth	The advantage of the system is that the author achieved better resource scheduling after clustering as a comparison to original resource distribution and better convergence speed as to make a comparison with the FCM algorithm
Rehman et al. [7]	2019	Min-Min algorithm	The advantage of the system is that Microgrid is provided close to the buildings and easily approachable by fog Cost, response time and execution times are the factors take care of by the author in the proposed work

(continued)

Table 1 (continued)

Approach	Year	Technique used	Advantage
Nazir and Shafiq [8]	2019	Cuckoo optimization algorithm	The advantage of the system is that cuckoo optimization works on updating utilized fogs and underutilized fogs to enhance the processing time of the system
Choudhari et al. [9]	2018	Prioritized task scheduling	The advantages of the system are reduced reaction time and decrease in cost of the suggested approach
Sun et al. [1]	2018	NSGA-2 algorithm Non-dominated sorting genetic algorithm MATLAB for simulations result	The advantage of the system is that the author uses a resource requester, if resources are required by edge-layer components, then the resource requester provides resources to the fog resource provider, which is any eligible fog node in a system in core cloud resource provider used, if the sufficient number of resources not provided by resource requester, then use of core layer has been used
Yin et al. [10]	2018	New virtualization technique based on container approach Docker 1.12.6 for container management	Advantages of the system are resource reallocation and scheduling the tasks

4. **Computing particle best position (pbest):** Particle current position value is pbest. If the newer pbest value is finer than the older pbest value, then the particle pbest value will be updated.
5. **Computing Global best (gbest):** Global best value is the finest value among all particles.
6. **Modernize:** Position vector and velocity vector value will be updated using the below equations:

$$Vi + 1 = \omega Vi + c1 \text{rand1} * (\text{pbest} - xi) + c2 \text{rand2} * (\text{gbest} - xi)$$

$$Xi + 1 = Xi + Vi + 1$$

where

- ω = inertia
- Acceleration coefficient = C1, C2

- Random numbers (rand1, rand2 which are uniformly distributed and $\epsilon [0, 1]$)
 - Particle best (pbest) = Particle best position value.
 - Global best (gbest) = amongst all population of particles, the best position value of particle.
 - i = repetition
7. Repeat steps II–VI until a stopping condition is met. The highest number of iteration may be the stopping condition or when successive iteration will have occurred, the fitness value of the particle feels no update.
8. **Output:** Final output will be the best particle as the solution will be print.

Algorithm Ant Colony Optimization:

1. **Loading:**
 - a. Positive constant will be assigned to pheromone value for every track in the middle of task and resource
 - b. Ideal solution = null
 - c. N number of ants will be placed on irregular material/resources
2. Each ant solution construction:
Repetition for every ant
 - (1) Beginning resource (material) will be placed for the first task of this ant in tabu list.
 - (2) Next for each left task
 - i. Choose for the t_i (next task) of next r_j (resource) for by implementing through transition rule
$$S_{ij} = \frac{(\tau_{ij})^\alpha (\eta_{ij})^\beta}{\sum_{k \in \text{allowed}} (\tau_{ik})^\alpha (\eta_{ik})^\beta}$$

if $j \in \text{allowed}$, allowed means not in the tabu list
Else 0
 - ii. Selected resources will be placed into the tabu list in the previous step of this ant End for Until solution of each ant will be built.
3. **Fitness Value:** The value of fitness of each ant will be computed from the fitness function.
4. **Replacement:** Optimal solution will be replaced with the ant's solutions, which have the best fitness value if its fitness value is higher than the ideal solution.
5. **Modernize the pheromone:**
 - a. Local pheromone will be modernized for each edge.
 - b. Global pheromone will be modernized for each edge.
6. **Clear each ant's tabu lists.**

7. Repeat steps II–VI until the stopping condition is satisfied. The highest number of iterations may be the or for consecutive iterations, there is no change in fitness value of ants' solutions.
8. **Output:** Print an optimal solution.

4 Experimental Results

The implementation of resource scheduling using PSO optimization algorithm is done on iFogSim Toolkit on the system having Intel core i3-2370M CPU processor @ 2.40 GHz and 4 GB RAM. The experimental result is showing a comparison of the implementation of the optimization algorithm on cloud and fog. Cost and makespan are enhanced on fog system for PSO optimization algorithm as the comparison with PSO and ACO on the cloud system.

Workflow scheduling comparison between the Fog and Cloud system is shown in the Table 2. The number of tasks that has to be implemented in the system is varying from 10 to 100 tasks. Minimize makespan time is shown in Fig. 2 as optimization algorithm PSO is implemented on the Fog system in comparison to PSO and ACO on cloud.

Workflow Scheduling comparison between the Fog and Cloud system is shown in the Table 3. The number of tasks that has to be implemented in the system is varying from 10 to 100 tasks. Minimize cost is shown in Fig. 3; as optimization algorithm PSO is implemented on the Fog system in comparison to PSO and ACO on cloud.

Table 2 Makespan time

Number of tasks	Makespan time		
	PSO-on fog	PSO-on cloud	ACO-on cloud
10	20.2	40.7	52.2
20	34.4	64.1	74.1
30	38.3	68.2	79.6
40	47.7	87.4	93.3
50	56.5	101.9	105.4
60	69.9	136.2	142.3
70	94.2	179.1	183.3
80	98.8	192.2	226.2
90	125.7	221.7	289.1
100	147.5	252.5	334.6

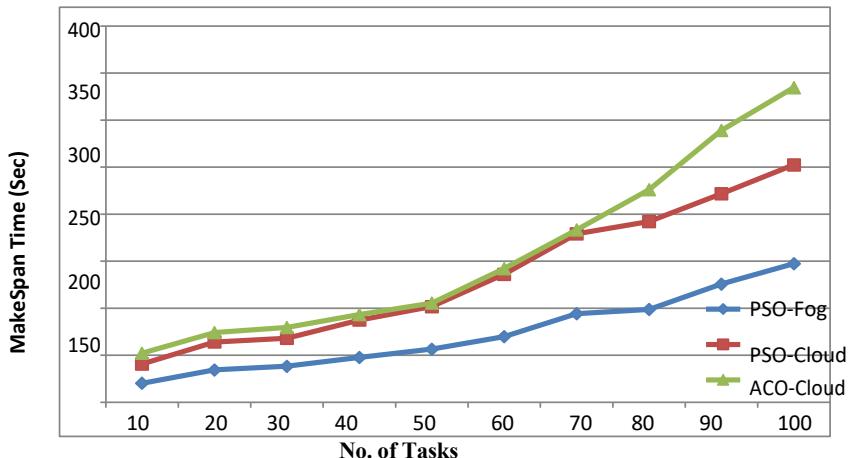


Fig. 2 Makespan time comparison

Table 3 Cost

Number of tasks	Cost		
	PSO-on fog	PSO-on cloud	ACO-on cloud
10	320	532	865
20	265	520	893
30	290	542	925
40	284	620	957
50	257	585	942
60	279	581	1017
70	245	618	1000
80	239	645	978
90	225	671	902
100	212	720	1130

5 Conclusion and Future Scope

In this paper, a technique of PSO (particle swarm optimization) is implemented on Fog environment with the different number of tasks ratio and made a comparison with existing PSO and ACO optimization algorithm, which was executed in the cloud environment. Based on two QoS parameters, makespan and cost, a minimization in the result was shown on both the parameters.

In the future, work can be extended to the implementation of algorithms on other QoS parameters like delay, and response time. And will do implement a hybrid optimization algorithm to achieve enhance and optimize results.

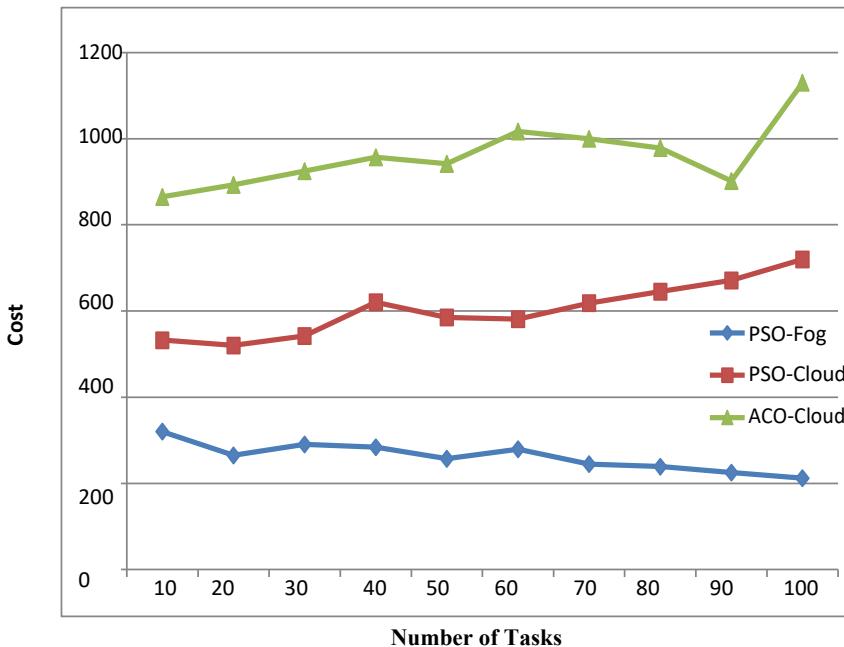


Fig. 3 Cost comparison

References

- Y. Sun, F. Lin, H. Xu, Multi-objective optimization of resource scheduling in fog computing using an improved NSGA-II. *Wirel. Pers. Commun.* **102**(2), 1369–1385 (2018)
- S. Bitam, S. Zeadally, A. Mellouk, Fog computing job scheduling optimization based on bees swarm. *Enterp. Inf. Syst.* **12**(4), 373–397 (2018)
- M. Ghobaei-Arani, A. Souri, F. Safara, M. Norouzi, An efficient task scheduling approach using moth-flame optimization algorithm for cyber-physical system applications in fog computing. *Trans. Emerg. Telecommun. Technol.* **31**(2), 1–14 (2020)
- M.K. Hussein, M.H. Mousa, Efficient task offloading for IoT-Based applications in fog computing using ant colony optimization. *IEEE Access* **8**, 37191–37201 (2020)
- A.M. Senthil Kumar, B. Kasireddi, An efficient task scheduling method in a cloud computing environment using firefly crow search algorithm (FF-CSA). *Int. J. Sci. Technol. Res.* **8**(12), 623–627 (2019)
- G. Li, Y. Liu, J. Wu, D. Lin, S. Zhao, Methods of resource scheduling based on optimized fuzzy clustering in fog computing. *Sensors (Switzerland)* **19**(9) (2019)
- S. Rehman, N. Javaid, S. Rasheed, *Min-Min Scheduling Algorithm for Efficient Resource Distribution Using Cloud and Fog in Smart Buildings*, vol. 25 (Springer International Publishing, 2019)
- S. Nazir, S. Shafiq, *Cuckoo Optimization Algorithm Based on Job Scheduling Using Cloud and Fog Computing in Smart Grid* (Springer Nature Switzerland, 2019), pp. 34–46
- T. Choudhari, M. Moh, T.S. Moh, Prioritized task scheduling in fog computing, in *Proceedings of the ACMSE 2018 Conference*, vol. 2018, January 2018
- L. Yin, J. Luo, H. Luo, Tasks scheduling and resource allocation in fog computing based on containers for smart manufacturing. *IEEE Trans. Ind. Inf.* **14**(10), 4712–4721 (2018)

11. N. Gao, C. Xu, X. Peng, H. Luo, W. Wu, G. Xie, Energy-efficient scheduling optimization for parallel applications on heterogeneous distributed systems. *J. Circuits Syst. Comput.* **29**(13), 1–28 (2020)
12. W.T. Vambe, C. Chang, K. Sibanda, A review of quality of service in fog computing for the internet of things. *Int. J. Fog Comput.* **3**(1), 22–40 (2019)
13. A. Wang, K. Batiha, A comprehensive study on managing strategies in the fog environments. 1–11 (2019)
14. M.A. Mohammed, I.A. Mohammed, R.A. Hasan, N. Tapus, A.H. Ali, O.A. Hammood, Green energy sources: issues and challenges, in *2019 18th RoEduNet Conference: Networking in Education and Research*, no. i (2019), pp. 1–8
15. M. Haghi Kashani, A.M. Rahmani, N. Jafari Navimipour, Quality of service-aware approaches in fog computing. *Int. J. Commun. Syst.* **33**(8), 1–34 (2020)
16. J. Pereira, L. Ricardo, M. Luís, C. Senna, S. Sargent, Assessing the reliability of fog computing for smart mobility applications in VANETs. *Futur. Gener. Comput. Syst.* **94**, 317–332 (2019)
17. A. Toor, N. Sohail, A. Akhunzada, J. Boudjadar, Energy and performance aware fog computing: a case of DVFS and green renewable energy. *Futur. Gener. Comput. Syst.* **101**, 1112–1121 (2019)
18. T.R. Gulista khan, Load balancing through distributed web caching with clusters, in *Proceeding of the CSNA* (2010), pp. 46–54
19. I. Sharma, R. Tiwari, A. Anand, Open source big data analytic technique, in *Proceedings of the International Conference on Data Engineering and Communication Technology* (Springer, Singapore, 2017), pp. 593–602
20. J. Luo, et al., Container-based fog computing architecture and energy-balancing scheduling algorithm for energy IoT. *Futur. Gener. Comput. Syst.* **97**, 50–60 (2019). 18
21. G. Lal, T. Goel, V. Tanwar, R. Tiwari, Performance tuning approach for cloud environment, in *The International Symposium on Intelligent Systems Technologies and Applications 2016 Sep 21* (Springer, Cham, 2016), pp. 317–326
22. R. Tiwari, N. Kumar, A novel hybrid approach for web caching, in *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing 2012 Jul 4* (IEEE, 2012), pp. 512–517
23. G. Peralta, P. Garrido, J. Bilbao, R. Agüero, P.M. Crespo, Fog to cloud and network coded based architecture: minimizing data download time for smart mobility, *Simul. Model. Pract. Theory* no. July, 102034 (2019). 19

Early Detection of Covid-19 Based on Preliminary Features Using Machine Learning Algorithms



Madhav Sharma, Ujjawal Prakash, Anshu Kumari, and Kanika Singla

Abstract In this paper, early detection of Coronavirus has been proposed using some machine learning techniques. Coronavirus has been the most exceptionally infectious and dangerous disease in the year 2020. By analyzing the significant clinical symptoms, early detection of this disease in patients can be done. A clinical dataset has been used for the classification of the disease using support vector machines, random forest algorithms, and neural networks. Neural network has been the most promising algorithm in providing the best-performance parameters like F1-score, recall, precision, confusion matrix, etc. Experimental analysis shows that neural network outperforms due to the approximation nature of the activation functions.

Keywords Covid · Covid-19 · Symptoms · Supervised learning · Support vector machine · Random forest · Artificial neural network

1 Introduction

An infectious disease, COVID-19, has spread across the globe. Back in late 2019, it was first seen in Wuhan, a city in China. WHO named COVID-19 as a “PANDEMIC” on 11 March 2020. Many countries banned international travel and till now it is banned in many countries. The World Trade is totally disturbed by this pandemic, and all the major economies went into recession [1]. The respiratory system gets

M. Sharma · U. Prakash · A. Kumari · K. Singla (✉)

Department of Computer Science Engineering, School of Engineering & Technology, Sharda University, Greater Noida, India

e-mail: kanika.singla@sharda.ac.in

M. Sharma

e-mail: 2018009090.madhav@ug.sharda.ac.in

U. Prakash

e-mail: 2018012300.ujjwal@ug.sharda.ac.in

A. Kumari

e-mail: 2018016097.anshu@ug.sharda.ac.in

most affected by this infection which can be fatal sometimes, especially in old-age group people [2].

The symptoms of COVID-19: fever, sore throat, chest pain, dry cough, diarrhea, etc. Millions of people have been infected by this virus including the children as well [3]. The side effects of Covid-19 are comparable to the indications of the common cold or flu. The size of the droplets may be typically between 5 and 10 microns. A lot of worldwide scientific community have been analyzing the effects of COVID-19, to foresee the economic situation, proper planning of hospitals and medical facilities, sociopolitical decision-making, etc.

As per the World Health Organization (WHO) data, COVID-19 or coronavirus causes respiratory illness and is spread through respiratory droplets and close distance contacts. These infectious droplets may potentially enter your body. In the present trend, the use of Machine Learning (ML) models has been leveraged to produce better results [4] as COVID-19 has an irresistible illness through the respiratory system.

In this paper, ML models [5] were implemented for the early detection of of COVID-19 based on preliminary features.

The paper has been divided into different sections which are as follows:

- Section 1: This section contains the introduction to the paper, aim, motivation, and objectives of the problem statement.
- Section 2: In this section, we discuss the related work or the literature survey of the concepts used during the research.
- Section 3: This section includes methods and tools used to achieve the objectives of research problem. Performance metrics and experimental analysis have been conducted in this section, respectively, for Support Vector Machines (SVMs), Random Forests (RFs), and Artificial Neural Networks (ANNs).
- Section 4: Results obtained have been shown in this section.
- Section 5: This section contains the analysis and discussions about the result, contribution of the work to the existing research. The possible future scope has also been discussed in this section.

2 Literature Survey

ML has a great capability of prediction and classification in healthcare systems [4, 5] using the forecasting mechanisms by applying an appropriate algorithm prediction of cases, number of deaths can be predicted which helps the healthcare institution to prepare well for the future and make the system more robust from earlier. Another line of work in the healthcare field using ML models [6] using SVM, RF, etc.

Other techniques like Linear Regression, Least Absolute Shrinkage and Selection Operator, and SVM [2] have been used in the study for forecasting the threatening factors of COVID-19.

Some state-of-the-art solutions are Artificial Intelligence (AI) and big data to fight against the virus. Generation of big data leads to a more accurate condition of the

world, by analyzing the big data and incorporating AI techniques such as ML, Deep Learning (DL), etc. The models can detect the COVID-positive patients easily [7].

Another work in this line of approach compares polynomial regression algorithm with SVM algorithm [8]. The former method showed an accuracy of approx. 93% by predicting the surge in cases for the months of July and August.

The intention of AI is to facilitate human limits for better results or output. Availability of clinical data AI is getting a standpoint on human administrations [9]. The need of AI is to fight help the world to fight against the COVID-19 crisis, also highlighting the application of big data. Methods used are like neural networks, SVM, and edge significant learning [10].

The most significant research in this area of work has been done by the authors [11]. In this research, epidemiological, demographic, clinical, laboratory, radiological, and treatment data from Zhongnan Hospital were analyzed. Radiological treatment has been given much leverage.

A new framework has been proposed for detection using inboard smartphone sensors. The framework will first collect the data from the sensors and predicts the infection of the disease [12].

In all the above-mentioned work, many prediction-based systems have been developed using a dataset containing images. Not much work has been done on the early detection of preliminary features of this infectious disease. So, this paper is focused on addressing this limitation. We will be using ML techniques to predict COVID-19 with clinical information of patients suffering from COVID-19.

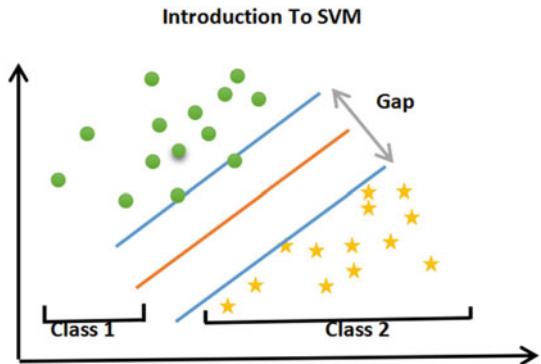
3 Methodology Used for Early Detection of Preliminary Features

3.1 Techniques

There is a number of classification models in ML like logistic regression, decision tree, neural network, SVM, Bayes classifier, etc. During this work, three supervised classification techniques have been performed and examined:

- A. **Support Vector Machines** SVM is one of the most popular supervised learning techniques which can be used for both regression and classification [8]. It has the concept of nonlinear kernels for creating the decision boundary for nonlinear data. Primarily, SVM-C is used for classification problems. The purpose of the SVM is to establish the boundary of the decision or the best line that can separate n-dimensional space into groups, which provides better intuition to calculate hinge losses between the hyperplanes [13]. Hyperplane is created with the help of SVM's chosen extreme points/vectors and these extreme cases are known as support a vector which leads to name this algorithm as Support Vector Machine [14] (Fig. 1).

Fig. 1 Support vector machine represented in a graphical way



- B. **Random Forest.** Like SVM, this technique can also be used for both regression and classification [15]. This technique is based on the basis of ensemble learning. Ensemble learning is a combination of multiple classifiers for solving a complex problem and for improving the performance of the model [8]. The random sampling and ensemble strategies [16] used in RF will help in predicting accurate and better generalization [17] (Fig. 2).

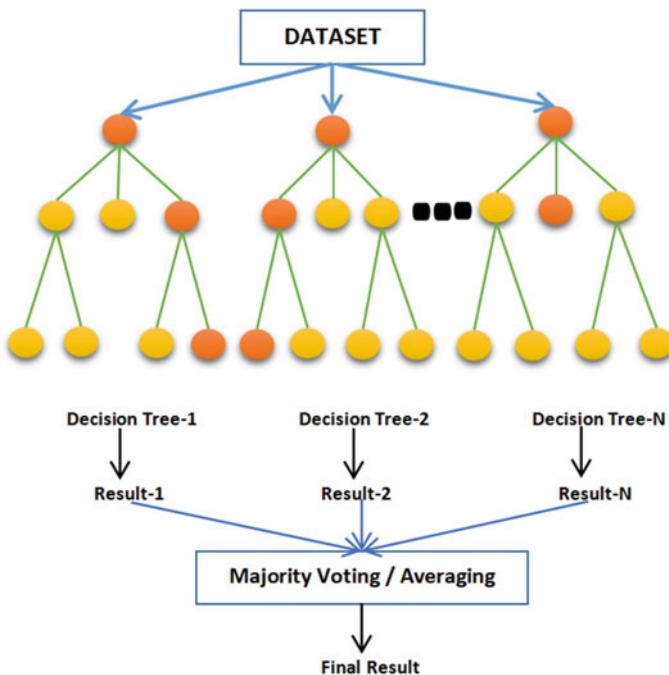


Fig. 2 Visualization of random forest techniques

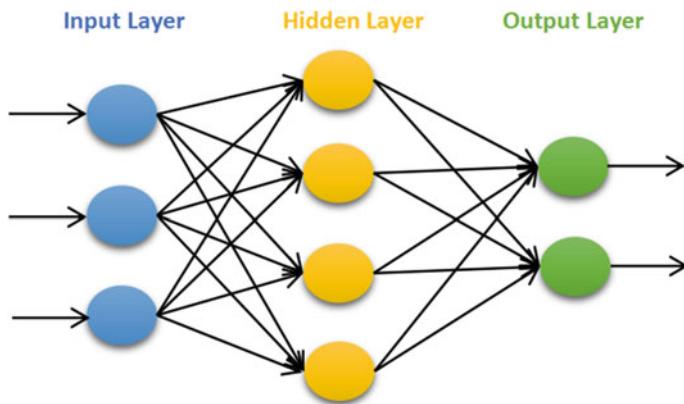


Fig. 3 Basic artificial neural network composition

C. **Artificial Neural Networks.** ANNs are used to mimic the human brain system [8]. It is the kind of framework used to analyze real-world data as the human brain does. It is the foundation of AI that helps in solving problems which seem impossible or hard for human brains to solve due to its complexity [18] (Fig. 3).

3.2 Dataset

The experiments were conducted on the clinical dataset. The dataset has been a public dataset [12]. The dataset contains information about hospitalized patients with COVID-19 (Table 1 and Fig. 4).

Table 1 Composition of dataset

Number of feature vectors	21
Number of observations/examples	5434
No of missing cells	0
Missing cells (%)	0.0%
Memory size	891.6 KB
Average record size in memory	168.0 B

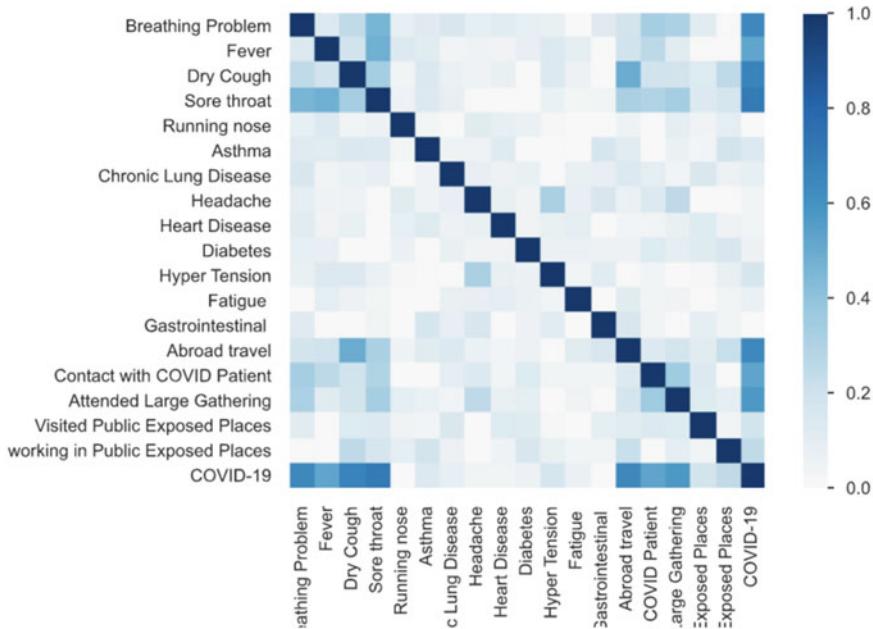


Fig. 4 Correlation between the feature vectors

3.3 Flow of the Work

Basic ML algorithms contain few basic steps as pipeline. In medical and healthcare applications, data processing is very crucial to get accurate results, as wrong predictions may lead to loss of human life. Extracting the essential features from the dataset before training the training of the model will give better results (Fig. 5).

3.4 Performance Metrics

- Accuracy.** Accuracy is used for the evaluation of classification models. For binary classification, accuracy can also be calculated in terms of positives and negatives:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (1)$$

where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

- Confusion Matrix.** The confusion matrix is a table frequently used to describe classification model's results (Fig. 6).

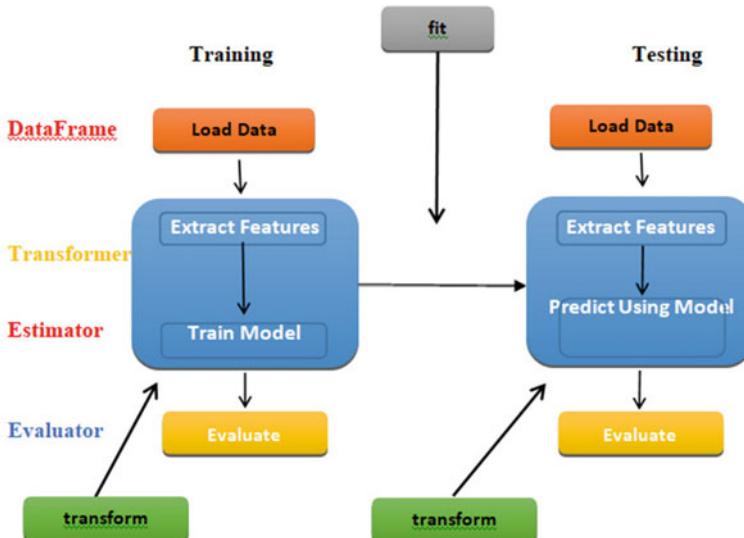


Fig. 5 Process of solving the problem

		True Class	
		Positive	Negative
Predicted Class	Negative	TP	FP
	Positive	FN	TN

Fig. 6 Confusion matrix

i. True Positive (TP)

If the value which has been estimated is same as the real value. The actual value was favorable; a positive value was also expected by the model.

ii. True Negative (TN)

When the predicted value comes to be same as the actual value. Actual value was negative; model was also predicted a negative value.

iii. False Positive (FP)

When predicted value comes to be falsely predicted. The model was predicted a positive value but the actual value was negative. It is also known as “**Type 1 error**.”

iv. **False Negative**

It is also known as Type-2 error. It is the error when model is estimated as false, and the actual result is also.

c. **Classification Report**

- i. **Precision:** A classifier's ability not to mark a positive example that is actually negative.

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (2)$$

- ii. **Recall:** The capacity of a classifier to classify all positive examples.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (3)$$

- iii. **F1-score:** Weighted harmonic mean of precision and recall. 1.0 is the best score and 0.0 is the worst.

$$\text{F1score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision}) \quad (4)$$

- iv. **Support:** The number of actual class occurrences in a dataset that is listed. In the training results, the imbalanced support could indicate systemic flaws in the classifier's recorded scores and may indicate the need for re-balancing.

4 Experiment Results and Discussions

This section presents the results that are obtained from the experiment. All the performance metrics that are explained in the above section have been utilized to evaluate the performance. All the methods mentioned in Sect. 3.1 have been exploited over this dataset.

4.1 Results Based on SVM and RF

The result section has been divided into two parts. This result section will give the result for SVM and RF.

The next result section will be based on neural network. Below is the table for comparison.

i. **Accuracy for SVM and RF**

See Table 2.

Table 2 Accuracy of SVM and RF

SVM	Random forest
97.79	98.16

ii. Confusion Matrix of SVM

See Table 3.

iii. Confusion Matrix of Random Forest

See Table 4.

iv. Classification report of SVM

See Table 5.

v. Classification report of Random Forest

See Table 6.

Table 3 Confusion matrix of SVM

		Positive	Negative	
Actual Values	Positive	TP = 179	FN = 13	Sensitivity = 0.93
	Negative	FP = 11	TN = 884	Specificity = 0.98
		Precision = 94.2	Negative prediction value = 0.98	Accuracy = 0.977

Table 4 Confusion matrix of random forest

		Positive	Negative	
Actual Values	Positive	TP = 184	FN = 8	Sensitivity = 0.95
	Negative	FP = 12	TN = 883	Specificity = 0.95
		Precision = 93.8	Negative prediction value = 0.99	Accuracy = 0.981

Table 5 Classification report of SVM

	Precision	Recall	F1-score	Support
0	0.94	0.93	0.94	192
1	0.99	0.99	0.99	895
Accuracy			0.98	1087
Macroavg	0.96	0.96	0.96	1087
Weighted-avg	0.98	0.98	0.98	1087

Table 6 Classification report of random forest

	Precision	Recall	F1-score	Support
0	0.94	0.96	0.95	192
1	0.99	0.99	0.99	895
Accuracy			0.96	1087
Macroavg	0.96	0.97	0.97	1087
Weighted-avg	0.98	0.98	0.98	1087

Table 7 Confusion matrix of ANN

		Positive	Negative	
Actual values	Positive	TP = 190	FN = 2	Sensitivity = 0.98
	Negative	FP = 13	TN = 882	Specificity = 0.98
		Precision = 93.5	Negative prediction value = 0.99	Accuracy = 0.987

Table 8 Classification report of ANN

	Precision	Recall	F1-score	Support
0	0.94	0.99	0.96	192
1	1	0.99	0.99	895
Accuracy			0.96	1087
Macroavg	0.96	0.97	0.98	1087
Weighted- avg	0.98	0.98	0.99	1087

4.2 Results Based on ANN Model

Due to the presence of ReLu activation function which actually helps in breaking the linearity of the data. **The accuracy is 98.62.**

i. Confusion Matrix

See Table 7.

ii. Classification Report

See Table 8.

5 Conclusion and Future Scope

In this paper, a well-organized literature work has been conducted for the existing algorithms for COVID-19 prediction and classification, but no such algorithm has

been found for the early detection of the preliminary features of the infectious disease. Therefore, few supervised algorithms like SVMs, RFs, and ANNs were implemented for the clinical dataset. These algorithms were trained on the said dataset. The outcomes show that RF accomplished maximum accuracy, i.e., 98.16 which surpassed SVM's accuracy, i.e., 97.79. But, implementing ANN model on the same dataset shows that "ANN model surpassed the ML algorithm's accuracy rate by achieving 98.62" which is even more accurate, which clearly shows the model based on ANN is giving accuracy more than ML-based models. Moving toward non-linearity, i.e., from ML algorithms to DL algorithms, we can observe more good results.

We hence believe that calibration of ensemble methods and DL models and architectures can provide better solutions to the complex datasets which are highly non-linear in nature. The future scope may include finding a more non-linear dataset for COVID-19. Moreover, to identify and help diagnose the disease, various AI-based applications can be developed.

References

1. F. Rustam et al., COVID-19 future forecasting using supervised machine learning models. *IEEE Access* **8**, 101489–101499 (2020). <https://doi.org/10.1109/ACCESS.2020.2997311>
2. E. Gambhir, R. Jain, A. Gupta, U. Tomer, Regression analysis of COVID-19 using machine learning algorithms, in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 2020, pp. 65–71. <https://doi.org/10.1109/ICOSEC49089.2020.9215356>
3. W. Liu, Q. Zhang, J. Chen, R. Xiang, H. Song, S. Shu, L. Chen, L. Liang, J. Zhou, L. You, P. Wu, Detection of Covid-19 in children in early January 2020 in Wuhan, China. *N. Engl. J. Med.* **382**(14), 1370–1371 (2020)
4. G. Vashisht, A.K. Jha, M. Jailia, Predicting diabetes using ML classification techniques, in *International Conference on Innovative Computing and Communications* (Springer, Singapore, 2021), pp. 845–854
5. V. Awatramani, D. Gupta, Parkinson's disease detection through visual deep learning, in *International Conference on Innovative Computing and Communications* (Springer, Singapore, 2021), pp. 963–972
6. P. Jain, C.A. Babu, S. Mohandoss, N. Anisham, S. Gadade, A. Srinivas, R. Mohan, A novel approach to classify cardiac arrhythmia using different machine learning techniques, in *International Conference on Innovative Computing and Communications* (Springer, Singapore, 2021), pp. 517–526
7. M. Jamshidi et al., Artificial intelligence and COVID-19: deep learning approaches for diagnosis and treatment. *IEEE Access* **8**, 109581–109595 (2020). <https://doi.org/10.1109/ACCESS.2020.3001973>
8. S. Jie, H. Wankun, Experimental results of maritime target detection based on SVM classifier, in *2020 IEEE 3rd International Conference on Information Communication and Signal Processing (ICICSP)*, Shanghai, China, 2020, pp. 179–182. <https://doi.org/10.1109/ICICSP50920.2020.9232038>
9. A.A. Hussain, O. Bouachir, F. Al-Turjman, M. Aloqaily, AI techniques for COVID-19. *IEEE Access* **8**, 128776–128795 (2020). <https://doi.org/10.1109/ACCESS.2020.3007939>

10. E. Casiraghi et al., Explainable machine learning for early assessment of COVID-19 risk prediction in emergency departments. *IEEE Access* **8**, 196299–196325 (2020). <https://doi.org/10.1109/ACCESS.2020.3034032>
11. H.S. Maghdid, K.Z. Ghafoor, A.S. Sadiq, K. Curran, K. Rabie, A novel ai-enabled framework to diagnose coronavirus covid 19 using smartphone embedded sensors: design study (2020). [arXiv:2003.07434](https://arxiv.org/abs/2003.07434)
12. “COVID-19 Symptoms Checker | Kaggle.” <https://www.kaggle.com/iamhungundji/covid19-symptoms-checker>. Accessed 13 August 2020
13. E.-S.M. El-Kenawy, A. Ibrahim, S. Mirjalili, M.M. Eid, S.E. Hussein, Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images. *IEEE Access* **8**, 179317–179335 (2020). <https://doi.org/10.1109/ACCESS.2020.3028012>
14. R.I.H. Ortiz, J.C.B. Barrera, K.M.B. Barrera, Analysis model of the most important factors in Covid-19 through data mining, descriptive statistics and random forest, in *2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, Ixtapa, Mexico, 2020, pp. 1–8. <https://doi.org/10.1109/ROPEC50909.2020.9258765>
15. M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, B. Scholkopf, Support vector machines. *IEEE Intell. Syst. Appl.* **13**(4), 18–28 (1998). <https://doi.org/10.1109/5254.708428>
16. Y. Qi, Random forest for bioinformatics, in *Ensemble Machine Learning* (Springer, Boston, MA, 2012), pp. 307–323
17. D. Wang, B. Hu, C. Hu, F. Zhu, X. Liu, J. Zhang, Z. Peng et al., Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus–infected pneumonia in Wuhan, China. *Jama* **323**(11), 1061–1069 (2020)
18. E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms: bagging, boosting, and variants. *Mach. Learn.* **36**(1–2), 105–139 (1999)

Detection of Green Belt Area Using Machine Learning Algorithms



Ashish Raj Mahato, Rakshit Luke Wilson, Sushmita Sahoo, and Kanika Singla

Abstract A green belt is a region of land around the city where construction growth is severely limited. These green areas can be around roads, highways, and several other man-made infrastructures. This green space plays a significant and beneficial role in improving the city's ecological climate, reducing the impact of urban heat islands, industrial pollution, and fuel emissions, and promoting the harmonious relationship between humans and nature. However, not much recent research has been done in this area of application of machine learning. The goal of our paper is to analyze these green zones surrounding different areas which include agricultural lands, forest lands, side roads, and other kinds of natural environment activities, and provide a result as to if these green areas are being taken care of. In this work, we exploit Support Vector Machine (SVM) and Random Forest Classifier as a part of machine learning algorithms for the said task. The dataset of the green areas has been exploited through these algorithms for further analysis. The illustrative result shows Random forest classifier outperforms the SVM by exhibiting 89.11% accuracy.

Keywords Green belts · Green area · Support vector machines · Random forest · Machine learning · Analysis · Detection · Deep learning

1 Introduction

With the advent of image processing and visualization technology, many algorithms have been developed for the processing of images and extracting the important features for better predicting or classifying the data [1, 2]. The significance of the

A. R. Mahato · R. L. Wilson · S. Sahoo · K. Singla (✉)
Sharda University, Greater Noida, U.P., India
e-mail: kanika.singla@sharda.ac.in

A. R. Mahato
e-mail: 2017010828.ashish@ug.sharda.ac.in

S. Sahoo
e-mail: 2017003624.sushmita@ug.sharda.ac.in

green belt is to maintain a balanced relationship between nature and humans. This green space plays a significant and beneficial role in improving the city's ecological climate, reducing the impact of urban heat islands, industrial pollution, and fuel emissions, and promoting the harmonious relationship between humans and nature. Numerous nations, organizations, and associations around the globe have ecological approaches set up. Thruway's advancement included enormous earthwork and transformation of land utilized in its development [3]. This necessity applies the information on ecological science whereby control must be made in saving normal assets, simultaneously continuing the need of the present and group of people yet to come.

According to the Indian Govt. report of Feb 1, 2020, the government of India has finalized a budget of Rs. 28,720 crore toward the horticulture department for maintaining the greenery in the respective states. Much money is spent on this by the government every year and minimum output is seen. So, detecting the green area or the belt is an important concern both from the financial or ecological aspect. And the most important point of concern is that not much of the work has not yet been implemented in this application of machine learning [4].

The remote sensing technology has broken through the constraint of conventional methods and can remove green space rapidly and automatically [5], which provides the basis for urban greenbelt inventory and urban planning.

The government also suggests that the use of land in green belts has a constructive role to play in achieving a range of goals:

- Providing outdoor activities and outdoor leisure opportunities close to urban areas
- Retaining beautiful habitats and developing landscapes close to where people live
- To protect interests in the conservation of nature
- In agricultural, forestry, and related applications, the retention of land.

This paper focuses on the leverage of rule-based Random Forest (RF) classifiers and the Support Vector Machine (SVM) as the machine learning algorithms.

The paper has been divided into different sections which are as follows:

- **Section 1:** This section contains the introduction to the paper, aim, motivation, and objectives of the problem statement.
- **Section 2:** In this section, we discuss the related work or the literature survey of the concepts used during the research.
- **Section 3:** This section includes methods, tools used to achieve objectives of the research problem. Performance metrics and experimental analysis has been conducted in this section, respectively, for SVM, RFs, and Artificial Neural Networks (ANNs).
- **Section 4:** Results obtained has been shown in this section.
- **Section 5:** This section contains the analysis and discussions about the result, contribution of the work to the existing research. The possible future scope has also been discussed in this section.

2 Literature Survey

In previous years, efforts have been made to make better use of technology to conserve our surrounding environment. The use of technology which seems to be a bane to the developing world can be put to some great use for environmental conservation techniques. One such effort was made by a group of researchers in Mexico by developing ClasLite [6]. ClasLite, a forest monitoring system makes use of satellite images and remote sensing tools to detect disturbances in timber harvesting in four different regions. The team made use of unsupervised land classification methods and optical remote image sensing which provided an accuracy of 79.1% during the testing stages.

Another paper published by a group of researchers in China talking about the study on urban green spaces. They extended their efforts by extracting LANDSAT™ satellite images and making the use of decision tree classifiers and basics of SVMs to analyze the greenery of a very specific region. The use of SVMs increased the percentage of accuracy by 15% [7]. The researchers made use of the fuzzy-C method in the extraction of green spaces to solve the problem of mixed pixels in the imagery obtained from the LANDSAT satellite images. The overall accuracy obtained from the images in the year 2010 was 96.55%.

The extensive use of SVMs that provided efficient accuracies were inspiring. Another line of work has been dedicated to the conservation of the environment made use of high-density imagery, 3D detection, and 3D reconstruction to reconstruct a panoramic model and extract plan community structures. The remote sensing technology of an unmanned aerial vehicle played a crucial role to accomplish this effort. The reconstruction and remodeling of the images taken in by UAV provided an opportunity to analyze study and research more on the conservation practices [8].

Other techniques like Linear Regression, Least Absolute Shrinkage and Selection Operator, and SVM [5] have been used in the study for forecasting the threatening factors of COVID-19.

Another line of work [9] proposed the use of SVMs for Forgery image detection. The proposed work shows the measured performances of the AUC by the sensitivity (TP: True Positive rate) and 1-specificity (FP: False Positive rate) is above 0.9.

3 Methodology Used for Detection of Green Belt Area

3.1 Techniques

There are a number of classification models in machine learning like logistic regression, decision tree, neural network, SVM, Bayes classifier, etc. During this work, two supervised classification techniques have been performed and examined:

- A. **Random Forest Classifier:** RF classifier is a machine learning algorithm that makes the use of different data samples and then further makes the use of

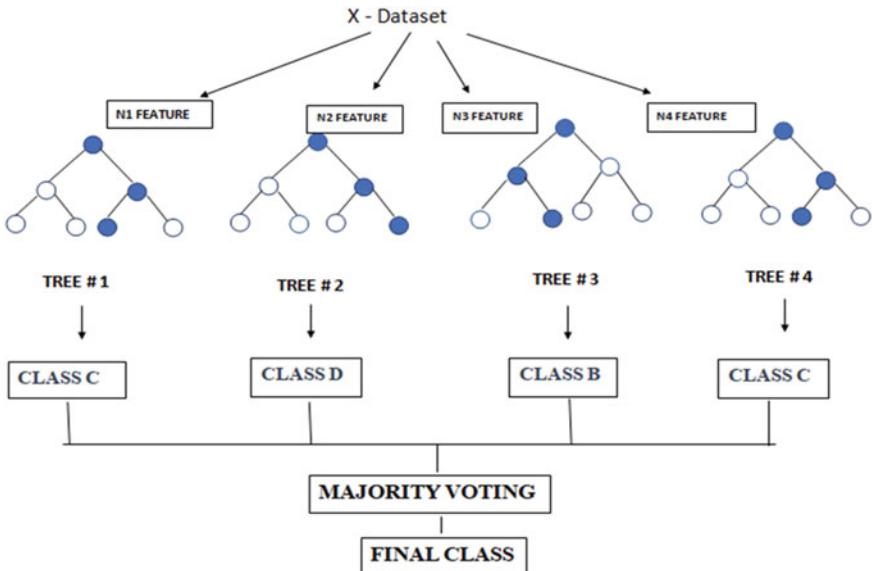


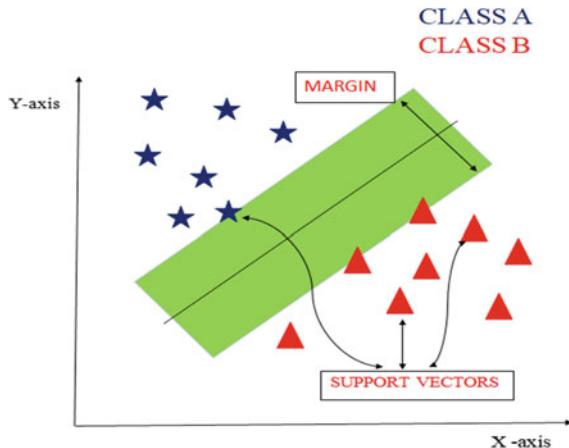
Fig. 1 Visualization of random forest techniques

voting of these samples to get essential results [5]. As the name suggests, like a robust forest is made up of multiple trees, the more the number of trees the better the forest is. The RF classifier creates a decision tree for each of the data samples and analyzes it. These decision trees are then put up in a process of voting for the predicted decision trees. The most preferred decision tree is put out after this classification and prediction process. RF classifiers are overfitting by averaging the results from these data samples. The algorithm can be used not only for classification but also for regressions but seems more effective for classification problems [10] (Fig. 1).

- B. **Support Vector Machine:** SVM is a supervised machine learning algorithm that makes the use of different data samples on an n-dimensional plane and helps in making classification of the data samples by interrupting the data with a hyperplane. This hyperplane acts as the essential element to classify these data samples [11]. The extreme points toward the hyperplane in the n-dimensional plane separated by a hyperplane are known as support vectors. These support vectors might affect the position of the hyperplane further categorizing SVM in linear SVM and non-linear SVM. Like the RF classifier algorithm, the SVM algorithm can also be used for regression analysis but is primarily used for classification problems [12, 13] (Fig. 2).

We consider that the problems of maintaining a clean and green city can be solved at a minimal budget if a proper check is kept on the already existing green life.

Fig. 2 Support vector machine represented in a graphical way



3.2 Dataset

The readiness datasets are imbalanced. We gave each classifier a shot of each dataset.

Augmented Dataset: For the **training of the model, a good amount of dataset is required, so data augmentation has been used for this purpose.** The dataset that we are utilizing for our calculation is fundamentally taken from a legitimate UK Govt. site which contains absolute 168+ passages of urban communities with their complete territories in hectare and followed by the green region of that specific city. We have gone through a process called data augmentation for our present dataset because the amount of dataset, we require for our algorithm is not available on the internet, so we have to go through this process called data augmentation in which we have added one particular digit to every entry possible and then doubled the given entries so that we get all the double number of entries that are already present on the internet.

The dataset is available in [14]; these statistics provide an update on Green Belt boundary changes and the area of Green Belt land in England from 1 April 2018 to 31 March 2019.

3.3 Flow of the Work

Basic machine learning algorithms contain few basic steps as pipeline. In medical and healthcare applications, data processing is very crucial to get accurate results, as wrong predictions may lead to loss of human life. Extracting the essential features from the dataset before training the training of the model will give better results.

I. Collect Data

We have gathered our dataset collected from a UK-based website which contains more than 168 entries of the city. The dataset has three significant segments that show the name of the city, its absolute assigned territory, and all-out green regions of that city. So, we have added one more column with a class classification which implies if the line is in Class (1), that city has great vegetation or that zone goes under green belt. Though if any line goes under Class (2) it implies the vegetation needs to be taken care of and the area is not under green belt. But, for now, we are using a dataset which is already on the internet. We will use that present dataset in our algorithm to get the desired output and check which of the algorithms gives us the best result with maximum accuracy.

II. Data Augmentation

We have gone through a process called data augmentation for our present dataset because the amount of dataset, we require for our algorithm is not available on the internet, so we have to go through this process called data augmentation in which we have added one particular digit to every entry possible and then doubled the given entries so that we get all the double number of entries that are already present on the internet (Fig. 3).

III. Classifier Training

We broke down RF Classifier, SVM as the classifier for our data. We used an external library called scikit-learn to do AI in Python. To plan, we need to first scrutinize the data from the Comma-detached characteristics (CSV) file that we made in the past cycle. By then, it has been arranged using Scikit'stree.RandomForestClassifier() work for RF classifier, SVM.

IV. Classifier

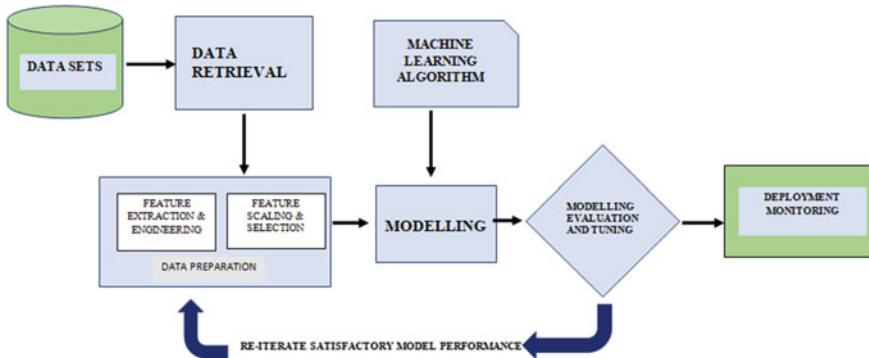


Fig. 3 Process of solving the problem

Exploration of different avenues regarding a couple of various sorts of classifiers: RF Classifier and SVM.

When utilizing the RF Classifier, we are utilizing the mean squared mistake (MSE) to how your information branches from every hub.

$$MSE = \frac{1}{N} \sum_{n=1}^N (x_i - y')^2 \quad (1)$$

where N is the quantity of information focuses, x_i is the worth returned by the model and y' is the real incentive for information point i .

When performing RFs dependent on grouping information, you should realize that you are regularly utilizing the Gini file, or the equation used to choose how hubs on a decision branch tree.

$$Gini = 1 - \sum_{i=1}^G (p_i)^2 \quad (2)$$

This equation utilizes the class and likelihood to decide the Gini of each branch on a hub, figuring out which of the branches is bound to happen. Here, p_i speaks to the general recurrence of the class you are seeing in the dataset and c speaks to the number of classes.

SVM attempts to limit base cost work subject to equity requirements rather than imbalance ones. Hence, an improvement issue in basic weight space can be given as

$$\min J(w, q) = \frac{1}{2} \omega^T \omega + \frac{1}{2} C \sum_{i=1}^N \xi_i^2 \quad (3)$$

$$\text{s.t. } yi = w_j(qi) + p + qi, \quad 0, i = 1, 2, \dots, Z \quad (4)$$

where (\cdot) $i \phi x$ is a kernel function which maps the low-dimensional data into higher dimensional data. The cost work J is communicated as an aggregate squared fitting mistake and a regularization term. In (1), C is the positive genuine consistent called the regularization boundaries and b is the “bias” term [15].

By changing this equation into a double structure with Lagrange multipliers α_i , the following equation arrives:

$$L(w, b, x, a) = \frac{1}{2} \omega^T \omega + \frac{1}{2} C \sum_{i=1}^N \xi_i^2 - \sum_{i=1}^N \alpha_i [w_j(x_i) + b + xi - yi]$$

The conditions for optimality are given by [14].

$$\begin{aligned}\frac{dL}{d\omega} = 0 &\Rightarrow \omega \sum_{i=1}^N \alpha_i \varphi(x_i) \\ \frac{dL}{d\alpha_i} = 0 &\Rightarrow y = \omega \varphi(x_i) + b + \xi_i i = 1, \dots, N \\ \frac{dL}{db} = 0 &\Rightarrow \sum_{i=1}^N \alpha_i = 0 \\ \frac{dL}{d\xi_i} = 0 &\Rightarrow C \xi_i i = 1, \dots, N\end{aligned}$$

These conditions are like standard SVM optimality conditions and can be composed as the answer for the accompanying arrangement of linear equations after disposal of w and x .

4 Experiment Results and Discussions

This section presents the results that are obtained from the experiment. All the performance metrics explained in the above section have been utilized to evaluate the performance. All the methods mentioned in Sect. 3.1 have been exploited over this dataset.

The results show that the best algorithm for our present dataset is SVM. We have divided the dataset into two parts, i.e., testing part and training part. In the training dataset, RF has an accuracy of 85.26% and SVM accuracy is 87.94%, whereas in the testing part RF has 86.3% accuracy and in SVM it is 89.11%.

The amount of dataset we require is not available over the public domains. That is why, accuracy is below 90% but in future, if we increase the amount of dataset the result would be better with a higher accuracy level. This result shows that the augmentation process was very useful for our dataset because it increases the accuracy of training and testing datasets (Figs. 4 and 5).

5 Conclusion and Future Scope

In this paper, a well-organized literature work has been conducted for the existing algorithms in respect to green belt area, but not as such an algorithm exists that works on the detection of green belt area. Therefore, few supervised algorithms like Support

Fig. 4 Graph showing the training accuracy between SVM versus RFC

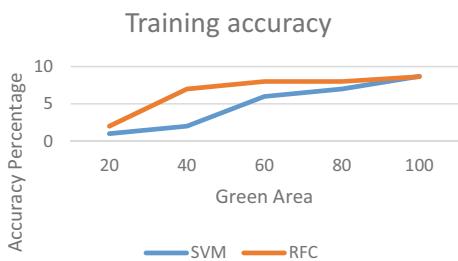
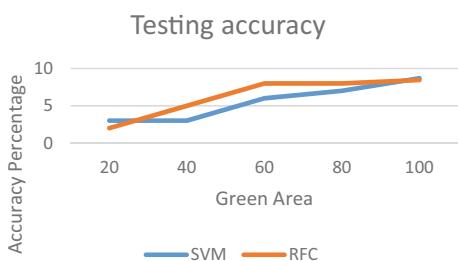


Fig. 5 Graph showing the testing accuracy between SVM versus RFC



Vector Machines (SVMs), Random Forests (RFs), and Artificial Neural Networks (ANNs) were implemented for the above-said datasets.

This work is a very impactful way to keep a check on the green belt on and around national and state highways. The use of testing and training data for this paper works as the key elements with the correct use of the verified data. The maintenance of green area is of at most importance but has been neglected in the past few days. Graphical results show that RF Classifier outperforms SVM.

The paper essentially requires the need for a verified and certified dataset. The requirement of a verified dataset is essential. The dataset currently worked upon is obtained from an official website governed by the government of the United Kingdom. The data obtained was not of the proper required amount to run it through a testing training phase, therefore, data augmentation on the dataset had to be applied. The use of testing and training splits for this paper works as the key elements with the correct use of the verified data. The maintenance of greenery is of at most importance but has been neglected in the past days and this work will help society with a solution for this problem with the step further in the right direction.

This paper currently works on the SVM and RF classifier algorithms and provides efficient results. It makes use of datasets obtained from certified and verified sources. The use of better datasets with a valid amount of data will in all likelihood improve the efficiency of the work. It will be our motive to implement and provide this existing work with more respectable and authenticated datasets in the future for better results.

Furthermore, we aim to work not only for numerical data but also take in use of images obtained from live sites for which the percentage greenery has to be verified. We consider CNN (convolutional neural network) as the best way to take a step

further in this direction. This major step in this project will claim it to be useful for the governments and individuals to keep a valid check on their surroundings and have a greener and fresher environment. This can be utilized to recognize the hazardous situation progressively by contrasting the ongoing information with authentic information. While the Green Belt Analyzer utilizes datasets consecutively coordinated into segments which are gotten from legislative or all-around authorized sources. Later, we will be utilizing CNN which will be better in classifying and improving the greenery for a particular sort of plain.

References

1. T. Karatekin, S. Sancak, G. Celik, S. Topcuoglu, G. Karatekin, P. Kirci, A. Okatan, Interpretable machine learning in healthcare through generalized additive model with pairwise interactions (GA2M): predicting severe retinopathy of prematurity (2019)
2. A. Lasisi, M.O. Sadiq, I. Balogun, A. Tunde-Lawal, N. Attoh, A boosted tree machine learning alternative to predictive evaluation of nondestructive concrete compressive strength. Okine (2019)
3. R. Farhat, Y. Mourali, M. Jemni, H. Ezzedine, An overview of machine learning technologies and their use in E-learning (2020)
4. Website showing the Guidelines and overall budget passed by Indian Govt. for green belt projects, <http://nhb.gov.in/guideline/stateandUt.pdf>
5. B. Babar, L.T. Luppino, T. Boström, S.N. Anfinsen, Random forest regression for improved mapping of solar irradiance at high latitudes. Solar Energy 198 (2020)
6. K. Rusek, J. Suárez-Varela, P. Almasan, P. Barlet-Ros, A. Cabellos-Aparicio, RouteNet: leveraging graph neural networks for network modeling and optimization in SDN. IEEE J. Select. Areas Commun. (2020)
7. W. Xinshuang, C. Erxue, L. Zengyuan, Y. Wanqiang, W. Lu, Study on urban green space extracting and dynamic monitoring method, in *2012 First International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*
8. T. Duan, L. Sang, P. Hu, R. Liu, L. Wang, Panoramic reconstruction of the central green belt of different levels of highway based on UAV platform, in *2019 5th International Conference on Transportation Information and Safety (ICTIS)*
9. K.H. Rhee, Forgery image detection of Gaussian filtering by support vector machine using edge characteristics, in *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, Milan, 2017, pp. 231–236. <https://doi.org/10.1109/ICUFN.2017.7993782>
10. W. Guohua, Y. Diping, Y. Jiayao, Z. Wenhua, D. Peng, X. Yiqing, Research on non-intrusive load monitoring based on random forest algorithm, in *2020 4th International Conference on Smart Grid and Smart Cities (ICSGSC)*
11. P.D. Windha Mega, Haryoko, Optimization of parameter support vector machine (SVM) using a genetic algorithm to review Go-Jek's services, in *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITSEE)*
12. M. Ross, C.A. Graves, J.W. Campbell, J.H. Kim, Using support vector machines to classify student attentiveness for the development of personalized learning systems (2018)
13. H. Elaidi, Y. Elhaddar, Z. Benabbou, H. Abbar, An idea of a clustering algorithm using support vector machines based on binary decision tree, in *2018 International Conference on Intelligent Systems and Computer Vision (ISCV)* (2018)
14. Dataset is available in the official website of UK government link is mentioned below, <https://www.gov.uk/government/collections/green-belt-statistics>

15. V. Calderaro, V. Galdi, A. Piccolo, P. Siano, A fuzzy controller for maximum energy extraction from variable speed wind power generation systems. *Electric Power Syst. Res.* **78**, 1109–1118 (2008)

Survey on Social Distancing Detection Using Deep Learning



Harsh Sandesara, Karan Shah, and Pramod Bide

Abstract This survey revolves around 8 research papers primarily focused on detection of Social Distancing using Deep Learning algorithms, for effectively combating Coronavirus (COVID-19). The main goal is to detect violation of social distancing practices. Several studies have been conducted and various innovative methods have been suggested. This paper aims to compare the proposed methods and their results. Till a vaccine is developed, social distancing is the only concrete way to control the spread of the disease. So, an accurate and feasible solution to detect social distancing is crucial.

Keywords COVID-19 · Social distancing detection · Tracking of humans · Object detection: you only look once (YOLO) · Region-based convolutional neural network (R-CNN) · Single shot detector (SSD)

1 Introduction

In early 2020, China witnessed an outbreak of 2019 novel coronavirus diseases. First reported in Wuhan, China, the virus quickly spread nationwide. And this disease didn't remain contained to just China; the outbreak rapidly expanded to many other countries. By March, 2020, in a matter of weeks, it had turned into a global pandemic [1].

H. Sandesara (✉) · K. Shah · P. Bide
Sardar Patel Institute of Technology, Bhavans Campus, Old D N Nagar,
Munshi Nagar, Andheri West, Mumbai 400058, Maharashtra, India
e-mail: harsh.sandesara@spit.ac.in

K. Shah
e-mail: karan.shah@spit.ac.in

P. Bide
e-mail: pramod_bide@spit.ac.in

The coronavirus could be either a mutated human virus which acquired new strengths, or an animal virus that can affect human cells, or a combination of two human viruses or a human and an animal virus. It has become a global emergency to develop safe, effective drugs and vaccines against the SARS-associated coronavirus as quickly as possible [2].

However, as yet, no vaccines or drugs have been found or developed that can completely kill or destroy the virus. The pandemic has now reached such a phase that it is considered the greatest public health hazard since the 1918 Influenza Pandemic. As a result, COVID-19 cases and casualties are growing exponentially. Hence, in the absence of vaccines, countries around the world are implementing the only way that could potentially slow down the spread of the virus—practicing various forms of “social distancing” [3].

Social distancing has become the chief measure taken by almost all the countries in order to curb the spread of COVID-19. The objective is to reduce physical contact between healthy individuals and people who could possibly be infected. According to a recent study, social distancing is one of the most important containment measures and is essential to prevent Severe Acute Respiratory Syndrome(SARS) CoV-2, because asymptomatic people or people with mild symptoms may also be able to transmit the disease [4]. According to the defined requirements by the World Health Organization (WHO), the minimum distance between individuals must be at least 6 ft. (1.8 m) in order to observe an adequate social distancing among the people [5].

A lot of technologies have been proposed for the detection of excessive crowding or contact tracing conduction, and communication plays a major role in most of them. Some examples of this communication are WiFi, Bluetooth, tracking based on cellular connectivity, Radio-Frequency Identification (RFID), Ultra Wide Band (UWB), etc. The problem is that, most of these technologies work well only indoors, though tracking pedestrians outdoors can be done by using cellular. Additionally, many of these technologies such as RFID, UWB, etc., require extra infrastructure or devices to track people indoors. In other cases, technologies like WiFi and Bluetooth are useful only for tracking the people who are connected to those technologies using wearable devices or smartphones. This causes limitations in their usage for tracking crowds and for monitoring the adherence of social distancing norms in general environments or public places, and may hinder the use of any kind of countermeasures [6].

Other viable methods often come under debates pertaining to privacy policy, as the users’ visual appearance or exact location path reveals more information than that which is needed for mere contact detection [7]. So, the system must be real time, without any capability for storing data [8].

Deep learning can be used to effectively monitor social distancing, keeping in mind the above points of concern. The paper contains 5 sections: Section 1 introduces the topic of the paper. Section 2 contains the literature survey. Section 3 compares the results of these papers and also contains a summarized comparison table of all the papers. Section 4 concludes the survey paper. Section V contains references for all the papers and any extra references used while writing the paper.

2 Literature Survey

Ever since the outbreak of coronavirus, social distancing has become the most important way of containing the spread of the virus. Monitoring this manually would take up a lot of time and energy, as well as introduce a margin for error. To counter this, various software and hardware solutions have been proposed. All these solutions require detecting the distance between individuals, and so the crux of it is object detection. Many sophisticated object detection algorithms have already been developed and so can be used directly to detect and find the position of people. There are various challenges in object detection, such as variation in clothes, postures, heights, distances and perspectives, and occlusion under different lighting conditions. Rezaei and Azarmi [5] proposed to develop their own human classifier and train their model based on a set of diverse datasets. Based on theoretical justifications and several experiments conducted by them, it was decided that CSPDarknet53 was the most optimal backbone model (extraction of features) for their application.

For the head model (class prediction and object location), they used the same configuration as YOLO (You Only Look Once)v3, which uses predefined boxes to detect multiple objects. The next phase of the proposed model was the tracking of people using SORT technique as a framework for the Kalman filter used to predict positions of humans at time $t + 1$ on the basis of current measurement at time t , with the Hungarian Optimisation technique to track people. This helped predict the overall track of each person, which was then used to estimate inter-person distances at different times by applying IMP to map 2D pixel points to corresponding world coordinate points, and categorizing each person into one of three categories: safe (green circle), high-risk (red circle), and potentially risky group of people moving together (yellow circle).

The paper also considered a spatio-temporal zone-based analysis over a long period, based on the movement pattern of people over that period, density of each zone, total number of people who violated the social distancing measures, the total time of the violations for each person and as a whole, identifying areas which are at greater risk, and ultimately, creating an informative risk heat-map.

A second paper, published by Singh Punn et al. [4], proposes a similar solution. The object detection model is trained and fine-tuned using YOLOv3 and Deepsort techniques, and generates a set of bounding boxes and an ID for each identified person. Each person is associated with a three-dimensional feature space and, by calculating the distance between people, each individual is assigned neighbors that satisfy the closeness sensitivity. The formation of groups shows that social distancing rules have been violated. This violation is quantified using a violation index.

Keniya and Mehendale [9] proposed their own self-developed model Social-distanci-ngNet-19 for detecting the frame of a person and classifying them as safe or unsafe depending on the distance between individuals. Their model has an architecture of 19 layers, where the input image is passed through a convolution, batch normalization and ReLU (Rectification Linear Unit) layers. Then it is passed through a single max pooling layer, two convolution layers, two batch normalization layers,

two ReLU layers, and a single addition layer. Then it is again passed through single convolution, batch normalization, and ReLU layers, and finally passed through a fully connected and a softmax layer. The feature extraction is carried out using a pre-trained CNN (Convolutional Neural Network) and they also used a reduced ResNet (Residual Neural Network)-50, MobileNet-V2, and ResNet-18 network. One caveat in this model is that, while using the webcam, detection goes incorrect if people stop moving continuously.

The above proposed applications might raise privacy issues because of continuous tracking of people. Also, they do not offer any solution to stop these violations. Going a step further than just detecting social distancing violations, a mobile robot using RGB-D camera and 2D lidar for collision-free navigation in a crowd was proposed by Sathyamoorthy et al. [6]. The robot was also equipped with a thermal camera to wirelessly transmit thermal images to healthcare personnel and could also use a CCTV (Closed-circuit television) camera (if available) in indoor setups to increase accuracy. It used Deep Reinforcement Learning (DRL)-based collision avoidance method and Frozone for minimizing occurrence of FRP (freezing robot problem). YOLOv3-based work was used for finding a set of bounding box coordinates for all pedestrians detected in the input RGB image. A breach of social distancing is detected when individuals maintain less than 6 ft. distance for a chosen 5 s threshold. After that, the robot navigates towards the location of the breach and encourages the non-compliant individuals to move away from each other through an alert message. It pursues them until they maintain the required distance. It also classifies non-compliant people in different groups and selects a goal which makes it move towards the vicinity of the largest group and enforce social distancing norms.

The drawbacks here were mainly caused by the limited FOV (Field of View) of the camera and thus improving that could provide better results. Also here, thermal images were transmitted to the personnel and they had to monitor the data continuously for checking if someone had higher than normal temperatures.

The system advocated by Yang et al. [8] contains a major focus on privacy rights of individuals. For this, they have made sure that no data is recorded/cached, no individuals are targeted in the warnings, no human supervisor should be present in the process, and the code should be open source, thereby being accessible to the public. They have used a fixed monocular camera to detect individuals in a ROI (Region of Interest) and measure the interpersonal distance in real-time. A pretrained state-of-the-art model such as Faster R-CNN (Region-based Convolutional Neural Network) and YOLOv4 is used for social distancing monitoring. If social distancing is breached, an omnidirectional audio-visual cue will be sent to warn the crowd. Also, a novel social density metric is proposed, so that people can be advised to not enter the ROI if the crowd density is higher than that value.

Kumbhar et al. [10] proposed a three-phase scheme to detect, track, and notify breach of social distancing practice. Video surveillance from CCTVs is fed to a fog node-based object detection model. The CNN-based model is trained with 2000 pictures marked with person class. Given a test image, the CNN based model produced a collection of all detected objects which calculated inter-person distance and identified violation immediately.

The paper also proposed identifying at-risk areas. The cellular devices in an area are monitored and scrutinized. Assuming that health centers have data of confirmed active cases, if a cellular user in the reported area is diagnosed as an active case, the health officials are notified, along with active users in the area. This requires them to self-isolate or contact a medical center. The active confirmed patient is quarantined.

The method also proposed the use of an IoT-based wearable device to check vitals of an individual. If a person shows symptoms of Coronavirus higher than the acceptable threshold, the surrounding people and health authorities are immediately notified, without revealing the identity of the affected person.

This method offers prevention, as well as detection of social distancing violations, and is implementable and scalable. However, the prevention of violation requires users to be continuously connected to the internet, which might be difficult to regulate.

Nadikattu et al. [11] have proposed a wearable hardware solution which makes use of PIR (Passive Infrared) sensors, microcontroller, and mobile for display and for giving alert to the user. The sensors will detect the distance between the wearer and other individuals by receiving infrared radiation from the human body. The range depends on the type and design of the sensor. To lengthen the detector's detection distance, an optical system can be added to receive the infrared radiation, using a flexible visual reflection system or a Fresnel lens. The simulation of this device was done using the Arduino-microcontroller, while the actual design was realized using Raspberry pie. Along with maintaining the required social distance, this device also gives alert whether a person in the range is having COVID-19 symptoms or not. A shortcoming of this approach is the PIR sensors cover a range of 240 degrees and not 360 degrees.

Bian et al. [7] proposed a wearable hardware solution to detect social distancing breaches. The device generates a magnetic field when switched on. The principal behind the device is to use Faraday's Law of induction to detect the presence of another similar device in the vicinity. The tests concluded that this solution provides a high accuracy of almost 100% within a range of 2 m, which is the acceptable range for social distancing. The devices and fields are also unaffected by everyday objects and are reliable even when there are obstacles present in the path of the devices.

Wearable hardware solutions provide an optimum solution for detecting and maintaining social distancing without needing video surveillance, with the only drawback being that they need to be worn at all times.

3 Result and Analysis

Various papers have used different pretraining models for object detection. The Single Shot Detection (SSD) method achieves the lowest accuracy of around 69.1% This method is one of the methods used in the proposed solutions of papers by Rezaei and Azarmi [5] and Singh Punn et al. [4].

Other methods used include YOLOv3, which achieves an accuracy of 84.6% when used in the systems proposed by Singh Punn et al. [4] and Mahdi Rezaei and Azarmi [5]; CNN, with an accuracy of 90%, used in the solutions proposed by Kumbhar et al. [10], and Faster R-CNN, with an accuracy of 96.9%, used in Singh Punn et al. [4].

The paper published by Kumbhar et al. [10] compared the two versions of YOLO (v2 and v3) and found that YOLO v3 outperforms YOLO v2 by achieving an mAP (Mean Average Precision) value of 90%, over YOLO v2's 73%. Yang et al. [8] experimented with two models for object detection: Faster R-CNN and YOLOv4. They were evaluated against Oxford/Mall/Train datasets and their mAP was 42.1%–42.7% and 41.2%–43.5% and inference time was 0.145/0.116/0.108 s and 0.048/0.050/0.050 s, respectively.

The paper published by Rezaei et al. [5] also tested accuracy using a DeepSocial method, and achieved a high accuracy of 99.5–99.8% for individual detection. However, couple detection was found to be around 98.7%, which is lower than normal human detection.

The accuracy of the self-developed model of Keniya and Mehendale [9], Social-distancingNet-19 was found to be 92.8%. They also found that the accuracy of the ResNet-50 network was 86.5% and that of ResNet-18 was 85.3%.

The hardware solution proposed by Bian et al. [7] demonstrated that social distancing detection was nearly 100% accurate when the distance between people was less than 2 m, which is the acceptable distance for social distancing.

The mobile robot solution presented by Sathyamoorthy et al. [6] had a high accuracy when pedestrians were closer to the robot. As they moved away from the robot and towards the extreme points of its camera's FOV, errors increased and a maximum error of 0.3 m was observed. The accuracy can be improved with higher FOV depth cameras. The robot-only configuration could detect upto 10 breaches, whereas the CCTV-only configuration could detect 20 breaches. The best performance was provided by the robot-CCTV hybrid combination. In static scenarios, 100% of the breaches were detected. It was also observed that the greater the maximum angular velocity of the robot, the better it could track a fast-moving pedestrian. Since the robot was navigating among humans, its angular and linear velocity was restricted to 0.75 rad/s and 0.75 m/s, respectively. This velocity capping made it challenging for the robot to track pedestrians walking at >0.75 m/s. This issue was again expected to be alleviated in the future when depth cameras improve their range and FOV.

3.1 Comparative Study of Social Distancing Detection Measures

Sr No	Methodology	Dataset used	Results	Gaps and issues
1	This paper proposes a 3-step model for keeping social distancing in check: detecting people from video footage, tracking their movements, and estimating the distance between people. The system can be integrated and used with all types of CCTV surveillance cameras with real-time performance [5]	Four common multi-object annotated datasets including Pascal VOC59, COCO60, Image Net ILSVRC76, and Google Open Images72, were tested for finding the number of bounding boxes for human or person. Testing of the proposed model has been done on the Oxford Town Centre (OTC) dataset	Accuracies: DeepSocial methods: 99.5–99.8%. YOLOv3: 84.6%. SSD: 69.1%. Coupled groups: 98.7%, recall-23.9fps (lower than normal human detection)	Since the model proposes detection using CCTV surveillance footage and is continuously active and tracking people, it might raise privacy and individual rights issues. The system only proposes to detect social distancing violations but does not provide any viable solution to avoid or rectify these violations
2	The paper proposes a deep learning-based framework that helps monitor social distancing using object detection and tracking models. It uses YOLO v3 and DeepSort for object detection. Based on the objects detected from the video feed and the distance measured among people, clusters of people who are in close proximity are formed, which indicates a breach of social distancing practice [4]	The discussed object detection models use the dataset acquired from the open image dataset (OID) repository maintained by the Google open source community. The images with a class label as Person are downloaded via OIDv4 toolkit along with the annotations	The YOLOv3 method achieved an accuracy of 84.6%. The SSD method was able to achieve results with an accuracy of 69.1%, while the Faster R-CNN method achieved 96.9% accuracy	Concerns about privacy and individual rights. The system only proposes to detect the social distancing violations, but does not provide any viable solution to avoid or rectify these violations

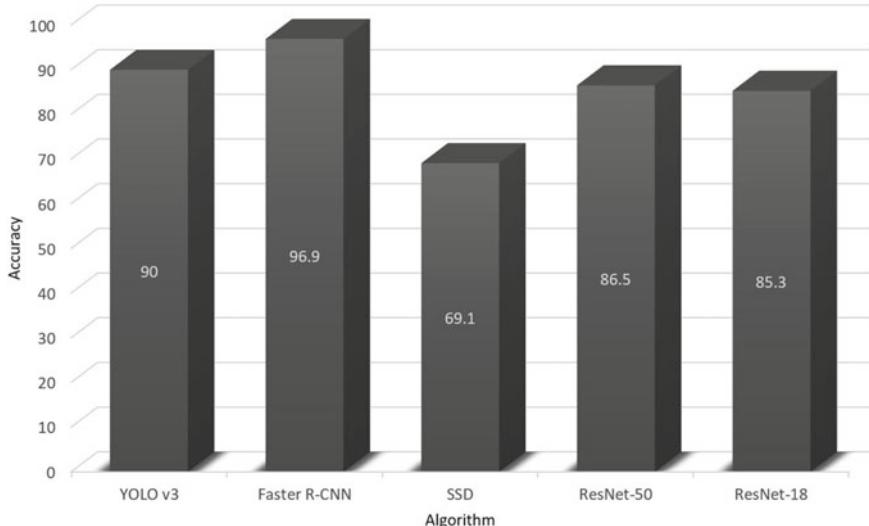
(continued)

Sr No	Methodology	Dataset used	Results	Gaps and issues
3	This paper proposes the use of wearable devices to detect and monitor social distancing. It has three major phases: 1. detecting social distance by analyzing video surveillance feed using YOLO v2 and v3 based on CNN 2. calculating the area of risk by scrutinizing confirmed disease cases and people having high symptoms of COVID-19 3. Personal tracing by tracking people who have been in contact with active confirmed patients or show high symptoms [10]	The training has an image database of 2,000 pictures, marked with person class. Each image has a resolution of 416×416 , with a batch size of 64 and the subdivision of 8	The YOLO v3 darknet-53 model based on CNN achieves 90% accuracy in object detection to identify inter-object distance and violation of social distancing. The YOLO v 3 outperforms YOLO v2 by achieving an mAP value of up to 90% as opposed to an mAP value of 73%	The proposed solution requires the user to wear an IoT (Internet of Things)-enabled device at all times, which might be inconvenient. The user needs to be continuously connected to the internet. Without a connection, the solution cannot be implemented successfully
4	This paper proposes the use of a mobile robot which uses an RGB-D camera, 2D lidar and for indoors, a CCTV-based setup if available. It uses DRL-based collision avoidance method along with Frozone for collision-free navigation in dense crowds. For detecting and tracking pedestrians, a YOLOv3-based scheme is used [6]	Raw data from 2D lidar, the robot's odometry, and the relative goal location	Errors increase as pedestrians go farther away from the robot. Maximum error is within 0.3 m and robot-CCTV hybrid combination provides best performance. 100% of breaches are detected in static scenarios	The robot is unable to pursue pedestrians which make a sharp turn and go outside its FOV. No detection of those who have higher than normal temperature and someone has to monitor data continuously. Also, no distinction is made between strangers and people from the same household

(continued)

Sr No	Methodology	Dataset used	Results	Gaps and issues
5	This paper focuses on privacy issues in social distancing detection, and hence proposes a non-intrusive AI-based active surveillance system for sending audio-visual cues in all directions, whenever a breach is detected. To detect individuals, a pretrained deep convolutional neural network is used with bounding boxes in a given monocular camera frame, measuring the interpersonal distance without recording data. It also measures social density and advises not to enter into the region if it is above a critical threshold [8]	A pretrained state-of-the-art pedestrian detector based on deep learning is used which has been evaluated against the datasets of Pascal VOC and MS COCO. Further, three case studies were conducted by them to evaluate the proposed method against the Oxford Town Center, Mall, and Train Station datasets	They were tested with two different deep CNN-based object detection models: Faster R-CNN and YOLOv4. Both achieved real-time performance. The mean average precision (mAP)% was found to be 42.1–42.7% for Faster R-CNN and 41.2–43.5% for YOLOv4 and the inference time recorded for Oxford Town Centre/Train Station/Mall datasets for them, respectively, was 0.145/0.116/0.108 and 0.048/0.050/0.050	Pedestrians who belonged to a group were not considered as a group in this method
6	They have used their own self-developed model named SocialdistancingNet-19 for detecting the frame of a person and displaying labels of safe and unsafe, monitoring people via video surveillance in CCTV. Their model has an architecture of 19 layers like convolution, ReLU(Rectification Linear Unit layers), and batch normalization layers. The feature extraction was carried out by a pretrained convolutional neural network (CNN) model. They also used a reduced ResNet-50, MobileNet-V2, and ResNet-18 network [9]	They loaded 295 images from the dataset, where each image had single or multiple labels inside it, which were used for training the model. Further, more images and labels were generated using an auxiliary dataset which was a variation of the images	The accuracy of the developed model SocialdistancingNet-19 was 92.8%. The accuracy of the ResNet-50 network was 86.5%. For ResNet-18, the accuracy was 85.3%	Privacy concerns due to video surveillance. Also, it was found that while using the webcam, it is necessary to have people moving continuously else the detection goes incorrect

Comparing the accuracies of various algorithms



4 Conclusion

This survey paper revolves around papers which provide a solution for social distancing detection and maintenance in the recent times of COVID-19. The main principle behind the solutions proposed is the detection of humans and calculating the distance between them to find possible breach in social distancing practices. A few papers also provide hardware solutions to prevent such violations and notify the concerned authorities about at-risk areas and patients. All this research helps monitor the current situation, and makes it easier to maintain control over the population and ensure that safety precautions are met. This compilation of different research papers can help an individual or a company decide the most applicable solution and aid them in selecting the correct application according to their wants and needs.

References

1. Epidemiology Working Group for NCIP Epidemic Response, Chinese Center for Disease Control and Prevention, The epidemiological characteristics of an outbreak of 2019 novel coronavirus diseases (COVID-19) in China. Zhonghua Liu Xing Bing Xue Za Zhi **41**(2), 145–151 (2020)
2. K.V. Holmes, SARS-associated coronavirus. N. Engl. J. Med. **348**, 1948–1951 (2003)

3. M. Greenstone, V. Nigam, Does social distancing matter? (March 30, 2020). University of Chicago, Becker Friedman Institute for Economics Working Paper No. 2020-26
4. N. Singh Punn, S.K. Sonbhadra, S. Agarwal, Monitoring covid-19 social distancing with person detection and tracking via fine-tuned yolo v3 and deepsort techniques (2020)
5. M. Rezaei, M. Azarmi, DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic, [arXiv:2008.11672](https://arxiv.org/abs/2008.11672)
6. A.J. Sathyamoorthy, U. Patel, Y.A. Savle, M. Paul, D. Manocha, COVID-Robot: Monitoring Social Distancing Constraints in Crowded Scenarios, <https://arxiv.org/abs/2008.06585v2>
7. S. Bian, B. Zhou, P. Lukowicz, Social distance monitor with a wearable magnetic field proximity sensor. Sensors **20**, 5101 (2020)
8. D. Yang, E. Yurtsever, V. Renganathan, K.A. Redmill, U. Ozguner, A vision-based social distancing and critical density detection system for covid-19 (2020)
9. R. Keniya, N. Mehendale, Real-Time Social Distancing Detector Using Socialdistancingnet-19 Deep Learning Network (August 7, 2020)
10. F.H. Kumbhar, S.A. Hassan, S.Y. Shin, New Normal: Cooperative Paradigm for Covid-19 Timely Detection and Containment using Internet of Things and Deep Learning, [arxiv:2008.12103](https://arxiv.org/abs/2008.12103)
11. R.R. Nadikattu, S.M. Mohammad, P. Whig, Novel economical social distancing smart device for Covid19. Int. J. Electr. Eng. Technol. **11**(4), 204–217 (2020)

Skin Burn Detection Using Machine Learning



Ashish Sharma

Abstract Skin burn identification is a very critical job to identify the burn location and its impact on the body. The current paper aims with the objective to identify the burn location and its impact so that the severity can be measured to provide effective treatment. The solution is derived using the Machine Learning model using CNN. The treatment can be provided after taking the right direction from the model. The relative features are identified and then based on the model the burn identification and its impact can be identified as well as we can measure the impact on it. The proposed approach based on CNN is tested on a standard burn data set of burns—BIP_US database. Training is done by classifying images into two groups. The test data set images are analyzed using the proposed CNN-based approach and 93% accuracy was achieved for the CNN-based model. The current method can perform better with a state-of-the-art machine learning technique on the burn images. Finally, COCO and BIS dataset is used to check the accuracy of the model. The final result illustrates the performance of the model, which is very effective in terms of accuracy.

Keywords Approximation clustering · Multi-SVM (support vector machine) · Burn detection · Machine learning · Deep learning

1 Introduction

A human body is encompassed by the skin and made sure about it. The skin gets hurt during consuming insults. The consume is one of the critical issues impacting open health. Consume consumes are extraordinary human issues—the fourth driving purpose behind consumes by death from unexpected injury [1]. The consumed body bit of the skin and close by tissues are affected in the consumption injury. In the wake of consuming, the essential treatment is required, with the chief to distinguish the size and level of the consumed part. Consume district, significance, and territory

A. Sharma (✉)

Department of Computer Engineering and Application, GLA University, Mathura 281406, UP, India

e-mail: ashishs.sharma@gla.ac.in

are essential factors in choosing to consume power. Subsequently, understanding the various sorts of skin and skin territory to choose the reality of the injury. It is made out of flexible fibers in the connective tissue. Conventionally, the three sorts of consumes are shallow dermal devouring, significant dermal expending, and full-thickness expending, and the separation between these three is the basic difference between these three sorts of burns [2]. The consumed consume parcel appears to be white or consumed. The basic piece of this investigation is to stamp such things with the objective that the most ideal thought can be given to them. The pictures of skin expending are improved also as master tests for fitting treatment, and they are then labeled [1]. The fundamental purpose of this work is to make a portrayal structure for the use of concealing qualities for pictures from a copy injury. The devouring of the skin may be isolated into three orders, for instance, shallow dermal expending, significant dermal devouring, and full-thickness devouring depending upon its shade concealing, concealing, and force, from the inside to the periphery. Shallow consumes recover from 14 to 21 days of genuine thought. It is basic to choose if a consume needs specific enthusiasm at the most prompt possibility. Deferred and wrong assurance can provoke extended threats for the affected individuals [2].

The ace definition incorporates taking out and joining mistook touches off for a cut. The brain to treat the entire piece of consume carefully with various conditions ensures effective assurance treatment. It needs an electronic structure to gather all the consumes as communicated previously. The survey may be finished at the join's base. As necessities seem to be, the accuracy normally changes with the readied consume master and is fresh from 78 to 93% [2]. The flow examines uses CNN strategies to cluster the images.

The paper is organized in the manner that Sect. 2 provides the related work, Sect. 3 describes the proposed model, Sect. 4 basic architecture of the model, Sect. 5 and 6 provide the result and discussion, respectively, and Sect. 7 contains the conclusion.

2 Related Work

Beginning late, there has been a surge in inevitable progress, for occasion, smart-watches and wellbeing trackers that can take after the human physical exercises without any issue. These contraptions have empowered commonplace tenants to take after their physical prosperity and encourage them to lead a solid way of life. Among different works out, strolling and running are the foremost broadly seen ones individuals do in standard ordinary nearness, either through the drive, work out, or doing family errands. At whatever point done at the correct control, strolling and running are satisfactorily palatable to assist persons with appearing up at the prosperity and weight diminish objectives.

Acah et al. [3] Fuzzy neural framework ARTMAP and SVM were used to control machines to find the duplicating significance. In any case, kurtosis offers a more definite duplicating significance than [3] since the proposed strategy can perceive touching off with different assessments. In this way, it is fundamental to degree

walking/running pace to survey the eaten up calories nearby keeping them from the hazard of disturbance, damage, and burnout. Existing wearable progress utilize GPS sensor to check the speed which is altogether imperativeness wasteful and doesn't work magnificently interior [4]. In the Tran et al. paper [4], the creators used the model to arrange consumed pictures and to portray the degrees of a copy. Appraisal of their consumes helps with recognizing the degrees of consumes and give subsequently. In this work, a strategy is proposed to see individuals and pets caught in eating up locales. This will be done such that it will guarantee the success of fire warriors and invigorate the protect procedure of mishaps. The proposed work proposes recognizing mishaps in fire conditions freely, through a noteworthy learning strategy utilizing the CNN demonstrate. The objective of the CNN progress is to hoard input pictures sent from the user location into one of three results classes: "individuals," "pets," or "no difficulties." The dataset was made to achieve the endeavor of human recognizable confirmation in Infrared (IR) pictures. The photos were set up to reflect high-temperature conditions as in developing on fire conditions [5, 6].

The central segment of expending is its power that is assessed by profundity. For this assessment, the prescribed course of action uses kurtosis to check the size of the consumes [7]. As regards consume conspicuous verification; first, the characteristics are picked for choosing decisions between solid skin and consumed skin. Another is working out the idea of the applications to make the device work. Tints expect a critical activity in picking the earnestness of a consume that the authority considers; this fuses the development of a zero-screw up picture plan system. In the related academic works, various procedures are proposed to help the consumer pro in choosing the right decision utilizing the automated program. Using concealing imagery helps with seeing the consumption wound's starting stage; it shows that the consumption will at first be taken rapidly to be against solid skin.

Rangaraju et al. [8] spread out a degree of consuming and joining procedure, a clinical strategy for choosing to consume structures. Because of weight on the skin, the hidden skin structure changes, and immovably stuffed material appears on the skin. Thickening (Coagulation) chooses the degree of consuming. In this paper, a promising learning-based methodology for programmed imperfection recognition dependent on little picture datasets. With the assistance of Wasserstein generative ill-disposed nets (WGANs), highlight extraction-based exchange learning strategies, and multi-model group system, our methodology can manage imbalanced and seriously uncommon pictures with deserts effectively, which is valuable to the assembling business [9, 10]. The author coordinated a relative assessment in Suvarna and Niranjan's paper on three orders [11, 12]. The proposed SVM-based model gives better results when it appeared differently concerning various techniques. This has moreover been shown to be the circumstance by picking the correct features in this work.

3 The Proposed Method

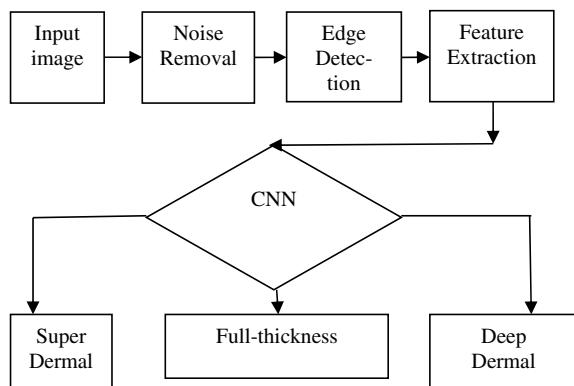
Techniques for perceiving the consumed skin as super dermal, significant dermal, and full-thickness wounds have been suggested in the literature [13–18]. Nevertheless, these frameworks need an identity, and can't perceive consume skin as being joining and non-joining together. The game plan of the join is considerably increasingly puzzled, and basic thought is required in view of its existence. Incidentally, non-join utilization is a ton of like ordinary expending and can be recovered quickly.

The suggested approach demands that the consume wounds to be inspected exclusively as excessively dermal, significant dermal, and full-thickness wounds. Skin joining framework is used to move or transplant skin to treat consumed skin beginning with one bit of the body then onto the following.

4 Model

Describing the consuming effect at first incorporates choosing the local condition of the consumed part, how much part is scorched. The ensuing conspicuous verification is required for concealing acknowledgment after the local shape has been set up. As the impression of human concealing gives consume type, an authority can without a very remarkable stretch separate the consume by its concealing. All tints in the range, for the most part, are open in concealing space, as tone outside the human experience. A channel implies estimation is required. In light of the site position, the mean estimation of the standard parts and the periphery segments move broadly. As showed by pros, too much dermal utilization doesn't require joining together. Be that as it may, it requires serious expending of the dermal and complete thickness. Along these lines, the educational assortment is isolated into three classes of Class-1, Super dermal, and Class-2 profound dermal, and Class 3 full thickness (Fig. 1).

Fig. 1 The CNN-based model for burn detection



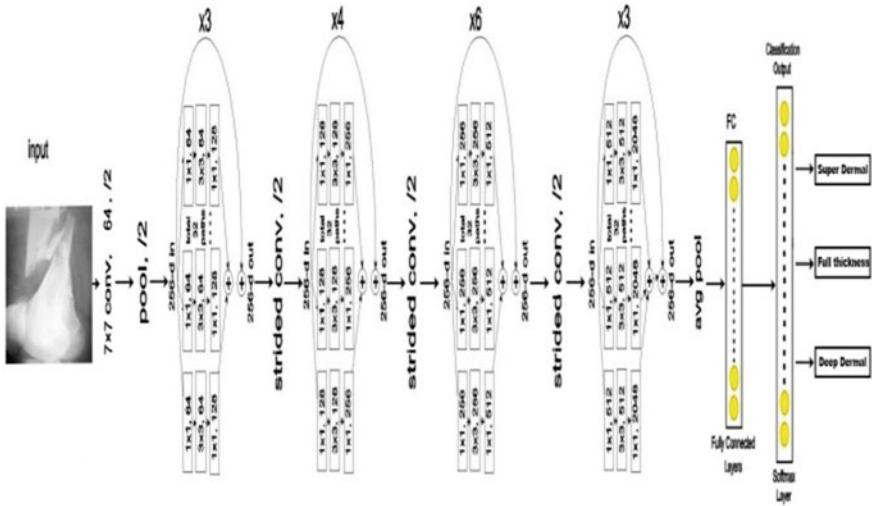


Fig. 2 The proposed CNN model

1. Augmentation: The size of the dataset is small, therefore, the overfitting may emerge [19]. It requires using some augmentation; the method is based on automatic augmentation technique to provide more and relevant data to perform the experiments.
2. Proposed Model: The CNN model basically contains convolution, pooling, smooth and large layers [6]. Figure 2 the presentation of each category is as follows:
3. CL: In the present experiment, different layers of convolution are used with the respective channel size. The convolution layer is responsible for the separating the element from the image information through channels.

5 Results

Data Set: The publicly available data set is taken from BIS US [20] and Common Objects in Context (COCO) data set. The test datasets which contain 74 jpeg images with size range from 3 to 78 kb.

The MATLAB version 16a on i7 processor with 8 GB of RAM is used in the execution of the programme.

6 Discussion

Describing Burn detection is done by CNN, the results are shown in Table 1. The

Table 1 Comparative result

S. no	Algorithm used	Accuracy (%)
1	MLSF-SVM [1]	84.12
2	EMS-SVM [2]	92
3	ARTMAP [3]	90
4	CNN-based model for burn	93

results are showing that the CNN is more effective approach as compare to traditional techniques like [1, 2]. The exhibition in the preparation procedure is 93 and 83.73%.

It is similarly fundamental to isolate the consuming picture to check the power of the light on the grounds that the qualification in light force gives a prevalent parcel, which is managed by the vacillation of, for instance, red channel assortment. The channel red vacillation is maximal during super dermal utilization and reduces a little bit at a time during significant dermal and most impressive thicknesses. The progression of the paper is that customized consume order is done on the component extraction utilizing CNN. The essential preferred position of the proposed approach is that the CNN gives better arrangements since it can without much of a stretch concentrate highlights from the consume pictures, as the ID of the component is so basic in this issue.

Experimental results are shown in Table 1 of test data sets which contain 74 jpeg images. The images are classified as superdermal, deep dermal, and full thickness.

The main advantage of the proposed method is that the function chosen for SVM classifier is capable of classifying the burn image into those that involve graft and non-graft. When photographs are recorded by different devices under different physical conditions, it becomes difficult to automatically analyze the pictures. The proposed method has, however, demonstrated greater precision than the previous method in accurately detecting and classifying the burn images.

7 Conclusion

Machine learning methods might be utilized to robotize the recognition of human burn images. The proposed strategy is a blend of extraction of features and a classification model that separates the imperativeness of the mathematical qualities and sees the consumption and its seriousness. In this way, the interest in computerizing the recognition of consumption is there. The proposed system isolated the highlights and recognized the consume and its earnestness. The CNN approach successfully recognizes the highlights along these lines, the exactness of the end and assessment of consumes ranges from arranged consume pros to natural burners from 76 to 93%.

In the methodology, the physical characteristics become intelligent features. In the hidden state of separation between whole skin and expending eat, the concealing imagery is used. Under a controlled circumstance, the learning system for masterminding consumes pictures is collected.

The exactness of the proposed approach is higher contrasted and the available writing in this examination, yet simultaneously ought to be watched out for a progressively critical collection of data. You may in like manner moreover improve the accuracy by picking explicit features, for instance, surface and concealing.

The accuracy of the proposed method in this work is higher compared with the available literature, but still needs to test on a larger data set. Also the accuracy could be further improved by selecting other features like texture and color.

References

1. A. Sharma, C.J. Varshney, D.P. Yadav, Sentiment analysis using ensemble classification technique, in *Proceedings of 2020 IEEE Students' Conference on Engineering and Systems (SCES)*, July 10–12, 2020, Prayagraj, India, Scopus Indexed. <https://doi.org/10.1109/SCES50439.2020.9236754>
2. A. Sharma, A. Mishra, A. Bansal, A. Bansal, Bone fractured detection using machine learning and digital geometry, in *Presented in the 1st International Conference on Mobile Radio Communications and 5G Network (MRCN-2020)* during 26th-28th March 2020, in University Institute of Engineering and Technology, Kurukshetra University, Kurukshetra, Haryana, India. Proceeding LNNS Springer, Scopus Indexed
3. B. Acha, C. Serrano, S. Palencia, J.J. Murillo, Classification of burn wounds using support vector machines, in *Medical Imaging 2004: Image Processing*, vol. 5370 (International Society for Optics and Photonics, 2004), pp. 1018–1026
4. D.P. Yadav, A. Sharma, M. Singh, A. Gupta, Feature extraction based machine learning for human burn diagnosis from burn images. *IEEE J. Transl. Eng. Health Med.* **7**, 1–7 (2017) Publisher IEEE Xplore. SCIE, PubMed, Scopus Indexed Journal (ISSN: 2168–2372). Impact Factor (2017): 2.075. <https://doi.org/10.1109/JTEHM.2019.2923628> (<https://ieeexplore.ieee.org/document/8766148>)
5. F.B. Jaradat, D. Valles, A victims detection approach for burning building sites using convolutional neural networks, in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)* (2020)
6. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016)
7. J. Hardwicke, R. Thomson, A. Bamford, N. Moiemen, A pilot evaluation study of high resolution digital thermal imaging in the assessment of burn depth. *Burns* **39**, 76–81 (2013)
8. F.B. Jaradat, D. Valles, A victims detection approach for burning building sites using convolutional neural networks, in *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)* (IEEE, 2020)
9. X. Le et al., A learning-based approach for surface defect detection using small image datasets. *Neurocomputing* (2020)
10. L.P. Rangaraju et al., Classification of burn injury using Raman spectroscopy and optical coherence tomography: an ex-vivo study on porcine skin. *Burns* (2018)
11. S. Mishra, R. Sagban, A. Yakoob et al., Swarm intelligence in anomaly detection systems: an overview. *Int. J. Comput. Appl.* 1–10 (2018). <https://doi.org/10.1080/1206212X.2018.1521895>
12. B. Sabeena, P. Raj Kumar, Diagnosis and detection of skin burn analysis segmentation in colour skin images. *Int. J. Adv. Res. Comput. Commun. Eng.* **6**(2) (2017). ISO 3297:2007 Certified
13. P.N. Kuan, S. Chua, E.B. Safawi, H.H. Wang, W. Tiong, A comparative study of the classification of skin burn depth in human. *JTEC* **9**(2) (2010)
14. R.M. Rangayyan, B. Acha, C. Serrano, *Color Image Processing with Biomedical Applications, Bellingham* (SPIE Press, USA, 2011)

15. V.D.R. Seethi, P. Bharti. CNN-based speed detection algorithm for walking and running using wrist-worn wearable sensors (2020), [arXiv:2006.02348](https://arxiv.org/abs/2006.02348)
16. M. Suvarna, U.C. Niranjan, Classification methods of skin burn images Int. J. Comput. Sci. Inf. Technol. **5**(1), 109 (2013)
17. X. Le, J. Mei, H. Zhang, B. Zhou, J. Xi, A learning-based approach for surface defect detection using small image datasets. Neurocomputing (2020)
18. H.S. Tran, T.H. Le, T.T. Nguyen, The degree of skin burns images recognition using convolutional neural network. Indian J. Sci. Technol. **9**(45) (2016)
19. C. Reischauer et al., Texture analysis of apparent diffusion coefficient maps for treatment response assessment in prostate cancer bone metastases—a pilot study. Eur. J. Radiol. **101**, 184–190 (2018)
20. Burns BIP_US database. http://personal.us.es/rboloix/Burns_BIP_US_database.zip [Biomedical Image Processing (BIP) Group from the Signal Theory and Communications Department (University of Seville, SPAIN) and Virgen del Rocío Hospital (Seville, SPAIN)]

Video Event Classification and Recognition Using AI and DNN



Sandeep Rathor, Nitika Garg, Prateek Verma, and Sarthak Agrawal

Abstract The paper proposes an efficient deep learning and AI approach for video event recognition. The proposed approach consists of several steps as video acquisition, framing, object detection, joint velocity detection, heat map generation, pose data collection, finally, the classification using deep neural network. The proposed approach is divided into two parts; the first part is object detection, in which we use bounding boxes to detect human presence. The second part is posing detection, which includes identifying human body joints by using open pose. Then, we extract features of body velocity; normalize joint positions and joint velocity. After this step we apply Principle Component Analysis (PCA) to reduce the features and then classify by using Deep Neural Network (DNN) of 3 layers of $50 \times 50 \times 50$. The proposed approach can render up to 6 fps; however, it can be improved with the GPU. The performance of the proposed approach is quite satisfactory with the accuracy of 94.67%.

Keywords Event recognition · Machine learning · Action recognition · Scene classification · Artificial intelligence

S. Rathor (✉) · N. Garg · P. Verma · S. Agrawal
GLA University, Mathura, India
e-mail: sandeep.rathor@gla.ac.in

N. Garg
e-mail: nitika.garg_cs16@gla.ac.in

P. Verma
e-mail: prateek.verma_cs16@gla.ac.in

S. Agrawal
e-mail: sarthak.agrawal_cs16@gla.ac.in

1 Introduction

Video is the most convenient means of communication. Various entities can be determined through the videos like emotion, feelings, activity, action, event, etc. The events can be defined by the interaction among different entities such humans and objects in the video [1, 2]. Event recognition is one of the most challenging research themes in the field of computer vision. Most of the efforts have been devoted to the problem of emotion and activity recognition from videos [3]. Images are static in nature therefore they are not much suitable for event recognition. However, we can recognize scene, mood, emotion through the images easily. Events are complex in nature to be recognized and depend upon several factors including human poses, facial expressions, garments, and object-scene contexts. Therefore object detection is first phase of event recognition.

In the recent years, the use of social media has increased exponentially thus increasing the amount of information shared through multimedia. Hence business analysts found a new way to gather the information of interest of their consumers. But the problem is most of the videos or images are captured by amateurs and hence causing disorientation in images, occlusion, and cluttered background making it difficult to extract useful information. Also most of the users are hesitant to annotate images and videos, thus giving rise to a big confrontation to the classical techniques of activities recognition that often cannot learn robust classifiers from a set of limited labeled videos and images of training.

Most activity recognition models use a conventional framework. Firstly, a large data is collected and then labeled through expensive human annotation. After this vigorous classifiers are learned from data collected for training. Finally, the presence of activities in an image or video is detected by the classifiers. Event recognition methods give promising results when sufficient and strong training labeled data set is provided. However, annotating a large number of data is time consuming and expensive. At present there are many successful and famous network architectures such as AlexNet, GoogleNet, VGGNet, ClarifaiNet, and many more. They settled to be significantly effective in object and scene detection and provided convincing results on various images and videos datasets. But their efficiency in event detection is disputable.

The formation of the proposed paper as follows related work in the same context has been discussed in Sect. 2. The proposed work in detail is described in Sect. 3. Section 4 discussed different datasets used to validate the proposed approach. Section 5 explains detailed results and finally, Sect. 6 describes conclusion and future scope of this approach.

2 Related Work

This section represents the work proposed by various researchers in the similar context. Activities can be classified into three categories as text-based activities, acoustic activities, and video activities.

The use of CNN to recognize activity in a scene has been proposed in [1]. In this paper author uses Object-Scene Convolutional Neural Networks to extract the important visual aspects of object as well as scene for understanding of activities. The OS-CNN is dissolved into two separate nets, namely, object net and scene net. However the problem with OS-CNN is that object net is outperformed by scene net for the detection of activities. Also there exists complementary property between object and scene nets for the detection of activities.

Visual activities recognition through videos was proposed in [3]. In this paper pyramid matching method ASPTM and A-MKL are used. It performs the fusing of information from different levels of pyramids and different types of local features. It also copes with the mismatch between the feature distributions of videos of consumers and web. However, the problem is that the author recognized the images and videos with very few labeled samples.

Complex activities recognition on static images by using fusing deep channels is proposed in [4]. This paper is inspired by deep learning technique and prepares a multi-layer framework which considers both visual appearance and interactions among entities such as humans and objects and fuses them via semantic fusion. However, there is a lot to do in detailed characterization of interactions, attributes of individuals, and the context. Object Detection and segmentation are proposed by [5]. In this paper an object detection method is proposed which combines top-down recognition with bottom-up image segmentation. This method consists of two main steps: a hypothesis generation step and a verification step. It is shown in the experiments that the simple framework is capable of getting both high recall and high precision even with few positive training examples. However, with the introduction of several Machine Learning Algorithms, it has become faded. High-Level event recognition in unconstrained videos is proposed by [6]. Key modules which are common are marked and provided detailed depiction along with notable insights for all of them individually, including extraction and representation of low-level features across multiple modalities, classification techniques, etc. But several issues like particular application requirements such as localization and recapping, along with scalability and efficiency are observed. Deep Sequential image features for acoustic scene classification are proposed by [7]. Author classified 15 different acoustic scenes using deep sequential learning, based on feature extracted from Short-Time Fourier Transform and scalogram of the audio scenes using Convolutional Neural Network. However, the best performance of this model is accomplished at different epochs and the performance can be upgraded using the pre-trained CNNs. On Semantics and Deep Learning for event detection in Crisis Situations are proposed by [8]. In this paper author, introduced a dual-CNN, an enhanced deep learning model in the state of crisis through the information provided by social media. A layer of semantics is added to CNN

because of the ill-structured data present in social media. However, in some tasks it was noted that models were overfitting even after the balancing of data and even after using semantic concepts of words in dual-CNN no significant improvement was found as compared to original CNN. The use of SVM-Hyper plane to recognize the augmentation in a scene is proposed by [9, 10]. The proposed framework generates feature samples using Generative Adversarial Networks (GANs) by using Support Vector Machine (SVM). However, for generalization of training of GAN generated data additional quantitative experiments are required. It is the drawback of this paper.

The main objective of the proposed paper is to provide an efficient approach for event recognition through video with the help of DNN. Our proposed approach works consistently on different video datasets.

3 Proposed Methodology

The proposed approach consists of three phases, i.e., object detection; pose estimation, and human tracking to recognize the activities. The detailed description is as follows:

(a) Object Detection

To estimate a human's presence in the image an algorithm capable of detecting human is to be used. There are the basic factors which decides the suitable selection of an object detection capable algorithm. Proposed object detection algorithm can be substituted by another such algorithm if the latter is efficient in terms of time and accuracy. The provided input into the object detector is in the form of RGB image [11, 12]. The scaling of inputted image can be adjusted for less time consumption in processing. The algorithm estimates the probable location of the object in the image in the form of bounding boxes.

(b) Pose Estimation

Post estimation approach is used for detection of human pose [13, 14]. It captures humans as objects and then an estimation of a human skeleton is done in this cropped region. A heat map is predicted through the pose recognition algorithms. Heat map consists of each joint probability which indicates the estimation of human skeletons along with the joints. At the stage of the post-processing it produces a 2-D human skeleton with its joint location.

(c) Human Tracking

The skeletal information of a human is necessary for recognizing events or activities in different frames. The human skeleton in different frames is absolute as the proposed algorithm is implemented on single static images. The connection of these frames is done through location of joints in the skeletons [15, 16] and consider the joint positions of the estimated skeletons in the frame as features which are to be recognized.

(d) Pose-Based Action Recognition

In the proposed model, we have investigated an approach of encoding human skeletons in form of data structure resembling images. This process is initiated by extracting the human pose from the web camera images. These extracted images are then encoded as joint positions using the x -, y - and z -coordinates, which represent joint positions as red, green, and blue. The joints used in this paper are nose, neck, hip center, both shoulders, both elbows, both wrists, both hips, both knees, both ankles, in a particular order. A $1 \times n \times 3$ matrix is used to assign the encoded joints in a particular order, where n denotes the number of joints. The encoded matrix is used in this process because it provides an image of Encoded Human Pose. To recognize the human activity we have used $m = 32$ to analyze one to two seconds of human movement. We have considered the recognitions of some frames and use of the action class with the highest summed probability in the previously considered frames in order to stabilize the human action recognition [17, 18].

Proposed Algorithm

The workflow of the algorithm begins with taking video as input to the model. Framing is performed on the input video. After the framing using the YOLO algorithm object detection is performed using bounding boxes. From this information the boxes containing human information are extracted. Then we get the joints' positions by Open Pose. It tracks each skeleton present in the frame. After this the Euclidean distance between the joints of two skeletons is used for matching two skeletons. It then fills in a person's missing joints by these joints' relative position in previous frame.

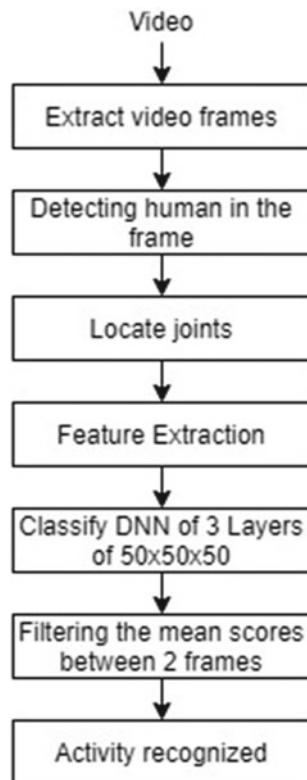
A window size of 0.5 s (5 frames) is used to extract the features. Extracted features are-(1) body velocity (2) normalized joint positions (3) joint velocities. Then Principle Component Analysis (PCA) [19] is used to reduce feature dimension to 80. Classification is performed by Deep Neural Network (DNN) of 3 layers of $50 \times 50 \times 50$. After this mean filtering is performed (Fig. 1).

Mean filtering predicts the scores between 2 frames. Then labels are added over the human image if the score is larger than 0.8. In this way the activity is recognized.

4 Dataset Used

The dataset is composed of 1528 videos from different standard video datasets like JHMDB HMDB52, UCF101, KTH which are classified into 21 action classes: brush hair, catch, clap, climb stairs, golf, jump, kick ball, pick, pour, pull-up, push, run, shoot ball, shoot bow, shoot gun, sit, stand, swing baseball, throw, walk, wave. Since there are 21 action classes in the dataset, we only need few action classes like run, walk, wave which can easily eminent from the pose data. The resolution of each video is 320×240 pixels. To validate our approach we have taken 60% videos of the dataset for the training purpose and 40% for testing purpose.

Fig. 1 Framework of the proposed approach



5 Result and Analysis

The proposed model's results are demonstrated through Figs. 2 and 3. Initially, when the frames are given as input to the model, the objects are detected through YOLO algorithm using bounding boxes to detect objects. YOLO's first step is to take an input image. This framework then divides the input image into various grids. Classification and localization are then applied on each grid. YOLO framework then predicts the bounding boxes and their class probabilities for objects. Then open pose is used to detect skeleton as shown in Fig. 2a. In the next step the coordinates of the body joints of a human are identified, shown in Fig. 2b. After this step a heat map is generated, which helps to identify the posture of the human body, shown in Fig. 2c, d. Finally Deep Neural Network (DNN) of $50 \times 50 \times 50$ layers is used to classify and label the human activity. Figure 2e shows the labeling of the activity Fig. 2c, d. Finally Deep Neural Network (DNN) of $50 \times 50 \times 50$ layers is used to classify and label the human activity. Figure 2e shows the labeling of the activity (Fig. 3).

The performance of the proposed approach on different datasets can be represented in Table 1.

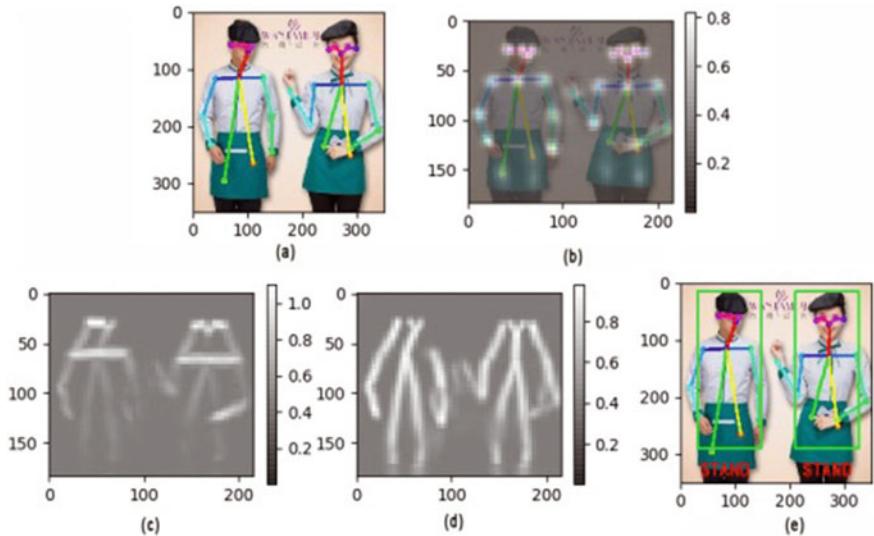


Fig. 2 **a** Skeleton detection, **b** joint detection, **c** heat map generation, **d** enhanced heat map generation, **e** labeling of the human activity

6 Conclusion and Future Scope

In this work, we have proposed a model that can recognize various activities done by humans in real time. In our proposed work human recognition is done by using object detection and activity detection with the help of Deep Neural Network (DNN) [23]. We have also compared the performance of our proposed approach on different video datasets like JHMDB, HMDB52, UCF101, KTH. Our approach on activities recognition works consistently on different datasets. For future point of view, we will include more number of activities.

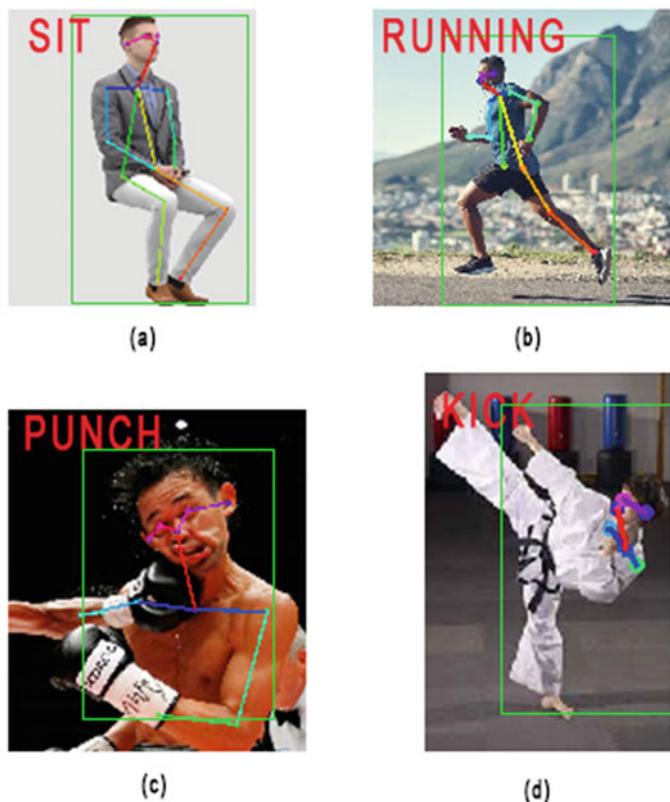


Fig. 3 Other human activities

Table 1 Accuracy of proposed approach on different video datasets

Data set name	Accuracy (%)
JHMDB [20]	93.8
HMDB52 [21]	95.2
UCF101 [22]	95.2
KTH [23]	93.9

References

1. L. Wang, Z. Wang, W. Du, Y. Qiao, Object-scene convolutional neural networks for event recognition in images, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2015), pp. 30–35
2. S. Rathor, R.S. Jadon, Acoustic domain classification and recognition through ensemble based multilevel classification. *J. Ambient. Humaniz. Comput.* **10**(9), 3617–3627 (2019)
3. L. Duan, D. Xu, I.W.H. Tsang, J. Luo, Visual event recognition in videos by learning from web data. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(9), 1667–1680 (2011)

4. Y. Xiong, K. Zhu, D. Lin, X. Tang, Recognize complex events from static images by fusing deep channels, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1600–1609
5. L. Wang, J. Shi, G. Song, I.F. Shen, Object detection combining recognition and segmentation, in *Asian Conference on Computer Vision* (Springer, Berlin, Heidelberg, 2007), pp. 189–199
6. Y.G. Jiang, S. Bhattacharya, S.F. Chang, M. Shah, High-level event recognition in unconstrained videos. *Int. J. Multimed. Inf. Retr.* **2**(2), 73–101 (2013)
7. Z. Ren, V. Pandit, K. Qian, Z. Yang, Z. Zhang, B. Schuller, Deep sequential image features on acoustic scene classification, in *Proceedings of the DCASE Workshop*, Munich, Germany (2017), pp. 113–117
8. G. Burel, H. Saif, M. Fernandez, H. Alani, On semantics and deep learning for event detection in crisis situations (2017)
9. S. Mun, S. Park, D.K. Han, H. Ko, Generative adversarial network based acoustic scene training set augmentation and selection using SVM hyper-plane, in *Proceedings of the DCASE* (2017), pp. 93–97
10. J. Redmon, A. Farhadi, Yolov3: an incremental improvement (2018), [arXiv:1804.02767](https://arxiv.org/abs/1804.02767)
11. J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2009), pp. 248–255
12. T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, C. Zitnick et al., Microsoft COCO: common objects in context, in *European Conference on Computer Vision* (Springer, Cham, 2014), pp. 740–755
13. B. Xiao, H. Wu, Y. Wei, Simple baselines for human pose estimation and tracking, in *Proceedings of the European Conference on Computer Vision (ECCV)* (2018), pp. 466–481
14. O.A. Alzubi, J.A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, M. Ramachandran, An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* (2020). <https://doi.org/10.1007/s00521-020-04761-6>
15. J.A. Alzubi, A. Kumar, O.A. Alzubi, R. Manikandan, Efficient approaches for prediction of brain tumor using machine learning techniques. *Indian J. Public Health Res. Dev.* **10**(2), 267–272 (2019)
16. D. Ludl, T. Gulde, C. Curio, Simple yet efficient real-time pose-based action recognition, in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)* (IEEE, 2019), pp. 581–588
17. H. Jhuang, J. Gall, S. Zuffi, C. Schmid, M.J. Black, Towards understanding action recognition, in *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 3192–3199
18. H. Hamdoun, S. Nazir, J.A. Alzubi, P. Laskot, O.A. Alzubi, Performance benefits of network coding for HEVC video communications in satellite networks. *Iranian J. Electr. Electron. Eng.* 1956–1956 (2020)
19. S. Rathor, S. Agrawal, A robust model for domain recognition of acoustic communication using bidirectional LSTM and deep neural network. *Neural Comput. Appl.* (2021). <https://doi.org/10.1007/s00521-020-05569-0>
20. C. Gu, C. Sun, D.A. Ross, C. Vondrick, C. Pantofaru, Y. Li, J. Malik et al., Ava: a video dataset of spatio-temporally localized atomic visual actions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 6047–6056
21. F. Afza, M.A. Khan, M. Sharif, S. Kadry, G. Manogaran, T. Saba, R. Damaševičius et al., A framework of human action recognition using length control features fusion and weighted entropy-variances based feature selection. *Image Vision Comput.* 104090 (2020)
22. K. Soomro, A.R. Zamir, M. Shah, UCF101: a dataset of 101 human actions classes from videos in the wild (2012), [arXiv:1212.0402](https://arxiv.org/abs/1212.0402)
23. Z. Gao, M.Y. Chen, A.G. Hauptmann, A. Cai, Comparing evaluation protocols on the KTH dataset, in *International Workshop on Human Behavior Understanding* (Springer, Berlin, Heidelberg, 2010), pp. 88–100

Systematic Survey on Cryptographic Methods Used for Key Management in Cloud Computing



Ramakrishna Oruganti and Prathamesh Churi 

Abstract Purpose Cloud computing is a new technology that aims to be the utility computing of the future. However, it is not without its share of threats and vulnerabilities. One of these threats is the security and privacy of the platform. One of the ways security is maintained is through cryptography, which includes the management of keys used for encryption and decryption. The data stored in the platform are encrypted, and the user must decrypt data when he needs it. This management is very important because once keys are compromised an attacker can hack and retrieve data very easily.

Methods The current problem in the key management of various encryption algorithms/frameworks is studied through existing literatures. For each method, the parameters like current key management methods, their limitations, and some new approaches to tackle the issue are listed and compared in the paper.

Results This survey paper gives an account of the current key management methods, their limitations, and some new approaches to tackle the issue. It also presents the analysis performed on some cryptographic algorithms, which are used in cloud computing. The paper also looks at the proposed frameworks for the platform and how they aim to mitigate the issues in cloud computing. Some of the new approaches are feasible and can be deployed into the cloud system, while some are not yet fully tested and require improvements.

Conclusion The paper gives a systematic review of various cryptographic methods, which are available for key management in cloud computing. The paper also gives a short overview of various evaluation parameters to be used to compare the existing cryptographic methods for key management in cloud computing.

Keywords Cloud computing · Cryptography · Security · Key management

R. Oruganti (✉)
Iowa State University, Ames, IA, USA

P. Churi
School of Technology Management and Engineering, NMIMS University, Mumbai, India
e-mail: Prathamesh.churi@ieee.org

1 Introduction

Cloud computing has become the most sought-after platform in modern computing. The simple concept, flexibility, and cost-saving solution have become very popular among organizations and individuals alike. The issue of security and privacy has become an important aspect of the platform. With the increase in popularity, the doubt of whether cloud computing is safe has become an important question on everyone's mind [1].

One of the most efficient and popular ways to ensure security is by using cryptography [2, 3]. Cryptography consists of two main components: encryption and decryption. Encryption converts plaintext into a cipher that must be decrypted for the data to make sense. This is performed by using keys one to encrypt and one to decrypt the data. Cryptography can be further divided into symmetric and asymmetric cryptography. Symmetric cryptography uses a single key for both encryption and decryption whereas asymmetric cryptography uses a pair of keys for each entity [4, 5].

For a cloud data center, sensitive data may be stored, and cloud providers cannot be trusted. The data can be stored over different virtual machines and users have minimal control over the data. The data are stored in encrypted form and there may be millions of users, everyone should not be able to access the data. There may be a group of users who access the same data and need a specific group key [3]. The role of key management is to enforce an access control on file systems and who can decrypt those files in the file systems. It supports the establishment and maintenance of key relationships between valid parties according to security policy. Precisely speaking, key management provides member identification and authentication, access control, generation, and distribution of key material. More attacks are aimed at the key management level; hence, key management is very important to avoid loss or theft of data [4]. The encryption algorithm, protocol, and implementation framework must meet the standards established such as the NIST encryption validation program. The security of key sharing and use is the core of the cloud data encryption paradigm and it must be managed centrally and meet cloud service interoperability requirements [2]. Group key management also faces the same issues. When a user joins or leaves a group, keys must be changed to ensure security is maintained. This is to prevent old users to decrypt new data to leak information [6, 7].

This paper discusses the current key management methods that are used by cloud platforms [2, 3, 8]. The limitations of these present methods and new novel approaches for the framework of key management are also discussed [4, 6, 8–10]. The present cryptographic algorithms used currently by cloud providers and their performance are also discussed [3]. The new approaches aim to improve on existing faults and limitations and attempt to provide a faster and secure platform.

2 Literature Survey

This section provides summaries of all papers surveyed.

2.1 A Survey of Key Management Service in Cloud [2]

The authors present three typical key management services in this paper. The services described are AWS CloudHSM, Keyless SSL, and STYX. The overview of each service is discussed, and the security of each service is analyzed.

- AWS cloud HSM: This is embedded in the AWS database and is managed by AWS. This service provides a hardware security module (HSM), which is a computer tool that processes cryptographic performance and provides secure storage of cryptographic keys. Users can perform a variety of tasks such as creating, storing, importing, exporting, and managing keys. They can use symmetric or asymmetric algorithms based on your preferences. HSM uses the user's privacy (VPC) cloud and the user can use standard VPC controls to manage HSMs. HSMs are isolated from other AWS networks and the administrator can manage HSM but cannot access user keys. Only the user can control the production and use of the keys, hence, the security level is high.
- Keyless SSL: The SSL key allows an organization to establish secure connections with users who use the on-site services. Using Keyless SSL, users do not stop accessing the private key when websites want to use cloud content delivery (CDN) infrastructure. This service uses TLS where clients and server have authentication certificates for each other. Cloudflare has its own internal certificate officer who signs and issues certificates to both affiliate organizations. There is strict control over how these certificates are issued and how the X.509 extension is used. The use of certificates provides high security and unauthorized organizations do not have access to the keys.
- STYX: This is a three-phase hierachal key management scheme and uses a hardware-assisted service acceleration for CDN applications. It is implemented based on Inter Software Guard Extension (SGX), Intel QuickAssist Technology (QAT), and SIGMA (SIGn-and-MAC) protocol. The performance is enhanced with QAT-based acceleration and provides a strict key security guarantee. STYX leverages Intel SGX primitives and security attestation and authority techniques. The keys are transmitted securely through authenticated SSL channels and each key is attested based on an anonymous attestation mechanism. Thus, keys are protected, and the security level is high and the cloud network is protected from malicious customers who migrate contents into it.

The paper provides a neat overview of current key management services in the cloud. The security performance of each service is discussed and is a useful tool for people who want to know if their cloud service is secure or not. It provides a simple

approach to the services offered by each management service and is very easy to read and understand.

2.2 Research on Key Management Infrastructure in Cloud Computing Environment [8]

Authors face a problem where a cloud provider has to make significant human growth. Currently, cloud services offer basic schemes and all security measures are in the hands of customers. Current agreements only protect sensitive information but not the main stores and key stores that also need to be protected as data. Access to supermarkets should be restricted to authorized organizations to help control access.

This paper suggests cloud key management (CKMI) infrastructure. This advanced CKMI is a collection of all policies, procedures, and technologies for managing all types of cryptographic keys in the cloud. Such infrastructure can lead to lower development costs and funding.

The proposed work consists mainly of two parts, cloud key management client (CKMC) and cloud key management server (CKMS). CKMS communicates with CKMC using the cloud key management interoperability protocol (CKMIP). CKMS communicates with the PKI equivalent key management system. CKMIP is a proposed protocol that establishes communication between cloud key management servers and cryptographic clients. The protocol is used for any cryptographic client from multi-owner use to cloud storage and is a comprehensive system built into the cloud system. This will help reduce operating costs and infrastructure while strengthening operational control and security management.

The protocol consists of three primary elements:

- Objects: These are symmetric and asymmetric keys, digital certificates for clients, and so on. Examples are certificates, private and public keys, policy templates
- Operations: These are the actions that can be performed on objects. They can include creating a key, generating a key pair, locate, derive, or destroy a key, validate a key, and so on. CKMIP uses both synchronous and asynchronous request/response model. The requestor can check the status of his operations
- Attributes: These are the properties of the object, such as the kind of the object, its ID, and so on.

The terms of the application are discussed in this proposed framework. CSMS manages cloud storage (CSN) to store data. CKMC exists in CSMS. When data require encryption, CKMC sends a request to KMS for the “CREATION” function. It pass the keyword that corresponds to the object symbols in its message. CKMS returns the object with its unique id. CKMC then requests “GET” to CKMS by forwarding a unique id, where CKMS returns the object and its symbols and key values so that cloud storage knows it is getting the right key.

This paper provides a way to manage the communication between cloud key management servers and cryptographic cloud clients. Such a framework will benefit if the cloud provider is reliable and non-hazardous. The draft is only proposed and there are no results or conclusions to prove that this framework will be provided in the paper. As future work, they can provide for the implementation of the protocol and compare it with existing agreements.

2.3 Key Management for Cloud Data Storage: Methods and Comparisons [3]

This paper provides a neat overview of current key management methods for cloud data storage and a comparison between those methods. The authors also applied the methods to various cloud environments such as public cloud and outsourced cloud. A comparison of symmetric encryption algorithms is presented. The limitations of the management methods are also discussed.

The key management methods discussed are presented in Table 1.

The authors present a comparison between symmetric algorithms AES, Triple DES, Blowfish, and RC4. The analysis was performed on varying file sizes, from

Table 1. Key management methods from research [3]

Key management at client side	Keys are maintained at the client side and data are stored in encrypted format at the server. It is scalable and secure
Key management at cloud service provider side	Keys are maintained at service provider side. If the key is lost, customer is unable to read data. It is scalable but not secure
Management of key at both sides	Key is divided into two parts and one part is each stored at client and server. Data can be decrypted only if keys are combined. If one part of key is lost, data cannot be recovered. This is also scalable and secure
Key splitting technique	Key is split and divided among users. To access data partial keys, need to be collected by the user. If k of n keys is combined, data can be decrypted. This is scalable and secure
Key management at a centralized server	This used an asymmetric key approach. Data are encrypted with public key and stored at server. To decrypt data, user uses his private key. If the central server is compromised, his method fails
Group key management	A group of users may access the same data; hence, a group key is generated. All users store part of the key. If someone joins or leaves, a new group key is generated

200 to 1000 KB. For encryption time, the results show that RC4 is the fastest among the chosen algorithms. Triple DES is the slowest. For decryption time, the results are similar RC4 being the fastest and Triple DES being the slowest.

The paper is very easy to read and provides a good comparison of the methods. The discussed limitations can help researchers understand the problem and work to mitigate the issues. The analysis of encryption algorithms provides a good measure of which algorithm works the best and can be used by researchers to work on new approaches.

2.4 A Framework for Key Management for Data Confidentiality in Cloud Environment [4]

This paper discusses the components in key management and the importance of each component. The authors then propose a new framework to explain the key management mechanisms. The aim of the paper is to protect the keys from attackers.

The components in key management mechanism and their functions are summarized in the Table 2.

The proposed framework includes a metadata table to protect data privacy. The metadata structure is managed by KMM. Metadata can include information such as key type, key status, and time. This is added to the information stored in the production key table. The user sends the request to the provider with the key, while the provider sends the request to KMM asking the key generator to generate a random

Table 2. Key management methods from research [4]

Key generation	Keys can be generated by the cryptographic algorithm or by auto random number generation. The KMM maintains a table containing information of the user, cloud service provider, and the time for generation of key
Key destruction	Keys, when are not in use, must be removed and destroyed in a secure manner. Keys must also be removed if they are not used within the stipulated period
Key establishment	This is done in three ways: key pre-distribution, key distribution, and key agreement. Pre-distribution distributes the key to users in reserve. Distribution occurs when keys are in demand and agreement is based on cost between user and provider
Key storage	Keys must be stored securely. Encryption is the most used technique and depends on the user how he wants to access the keys
Key change	Changing keys is an important aspect of any cryptographic system. Regular updates to keys are performed. When a key is compromised, it will cause a loss of data. Hence it is important to perform key changes when necessary
Key usage	This pertains to the frequency of usage of keys. As key usage decreases the frequency of key change increases. This happens when an attacker attempts to trace the keys. Symmetric keys must be changed with every use so data will not be accessible if the key is stolen or socially engineered

key. The key is then sent to the user and the format is stored on a table stored by the main management system.

Key management approach faces a number of challenges when attempting to manage and manage data encryption. Some of the challenges are preventing attacks such as invaders and evil intruders. As users increase, the demand for keys also increases and managing a large number of keys is a challenge. Cloud platform variability and data access are also some of the issues.

The framework proposed by the authors is a good solution. By using metadata, keys can be tracked in a better and efficient way. However, no results are presented to prove the claim. The inclusion of challenges faced provides a good insight to mitigate issues. This is a good paper clearly and concisely explaining key management issues.

2.5 A Framework for Secure Cryptographic Key Management Systems [6]

This paper discusses the problems associated with groups in cloud system. A group of users who access the same data need a group key. The group key management is the main topic of discussion in this paper. The authors propose a novel approach for efficient key management in multi-owner cloud environments.

The proposed mechanism allows authorized users to join, store, and share data files in a highly secured manner. The framework has multiple layers. The system is divided into multiple domains and each domain has a manager. The domain manager generates and distributes keys to all members in the group. The proposed system consists of the following entities:

- **Dynamic Manager:** These are domain managers that are chosen dynamically by data owners for a specified time interval. The dynamic managers are changed periodically. This will handle the key metadata to verify and track the behavior of the owner. The dynamic manager is like the domain manager and distributes keys to the group members.
- **Certificate Authority:** This is globally trusted entity that issues certificates to the entities. It maintains information about the data owner, domain managers, and issues group id to each group.
- **Data Owners:** These are the set of registered set of users who store their data in the cloud platform. Data owners define access policies and encrypt the data under the policy. Only users who have the decryption key can access the data.
- **Cloud and Cloud Service Provider:** This is the platform and resources provided by the service provider to the user to store their data.

The flow of the proposed scheme is presented. The data owner registers with the certificate authority. The CA verifies the owner and generates a secret key and group ID for each owner. The CA then randomly selects a dynamic manager for a predefined period and defines the frequency in which the dynamic manager will

change. Then the encryption algorithm encrypts the messages using Shamir secret sharing algorithm. Keys will be distributed based on assigned ID and each group member can be uniquely identified. To decrypt the data, users should a decryption key request to DM and CA. The Shamir secret sharing scheme is a technique to protect sensitive data. In this scheme, keys are divided into n parts and distributed over multiple entities. To generate key, the user must accumulate at least t out of n shares. Even if t-1 shares are compromised, no one can learn anything about the key with those shares. Hence, it is a very secure algorithm.

The authors present an analysis and provide a comparison with the existing methods. The simulation was performed with 50 users, 5 groups, and 5 different domain managers on a private cloud setup. The time for encryption and decryption is higher than the existing methods. The authors claim since the time taken is more the proposed it is more secure.

The paper provides a feasible solution for the problem of group key management. The concept is novel and can be implemented in present cloud systems. However, given the results, time taken is substantially more than existing methods and may become an issue in time-critical applications.

2.6 Cloud Computing Key Management Mechanism for Cloud Storage [1]

The authors introduce a new encryption method for important management systems. The new algorithm is based on multicast key management in the hypergraph to ensure privacy and confidentiality. The submitted algorithm uses an additional encryption scheme. The data owner must encrypt the data before uploading the data. The server must encrypt the data again before other users can download data. The proposed algorithm is used for group key management.

Member key relationships are defined as hypergraph $H = (U, E)$ where U is a member of the safe group and E is a collection of hyperedges. $E_i (K_i)$ is a group member of the group everyone in which you share Key Keys. The model uses the CKMC and CKMS models proposed in the paper [8]. The CKMS layer is used for distributing and updating CKMS keys. The cloud server connects directly with this layer. User layer is responsible for distributing and updating users' keys.

In file storage, the user requests cloud storage and the CKMS hyperedge where the user resides will generate an encryption key and file storage. User who uploads that file also provides the ID of other users who can download and view the file. This is kept in the access control list. The user must register and ask the owner of the file to gain access to the file. To download the file, the user sends the request details to the cloud server, cloud and encrypts the file with the group key that the CKMS hyperedge user is in. This is then sent to the user and the user must first unlock the key and proceed to encrypt the file. Buttons should be updated each time a user joins or leaves a group. This is done to ensure the security of the system.

The performance of the proposed scheme is presented. Due to over-encryption scheme used, data are encrypted twice, which adds an extra layer of security to the system. The cloud server also encrypts the data before users can download the data. This ensures data confidentiality and integrity. Since access control is restricted, everyone cannot access the file without the permission of the owner. The system is also scalable and suitable for large enterprises.

The paper provides a new approach for file storage using hyperedges in cryptography. There are no results presented in terms of encryption and decryption time. As future work, the authors can provide the encryption and decryption time and provide a comparison with the existing algorithms to verify the feasibility of this algorithm.

2.7 Management of Symmetric Cryptographic Keys in Cloud-Based Environment [9]

The authors aim to provide a secure management for keys on the cloud platform side. They discuss the limitation of current methods and propose a new method for cryptographic key management. They proposed the use of enhanced Shamir's algorithm. The new approach consists of a data splitter as the core of the framework. It is a protocol that will generate, store, distribute, and revoke symmetric cryptography keys as per user requirements.

Using the proposed protocol, user can generate a new symmetric key. This new key will be split into N parts using the new enhanced Shamir's algorithm. There will be the main piece K_n , which will be assigned to the consumer of the application. This partial key will be important as without this partial key, the original key cannot be recovered. The user will transfer the partial key using Public Key Cryptography Standard. The protocol used to send the keys will be signed by the sender, so authentication is also present. To recover the key, the client will collect partial keys from other clients. The original key will be computed on the fly with K_n . If this K_n is lost, key cannot be recovered.

The only difference between the original Shamir's and this enhanced algorithm is the presence of a main key piece K_n . The user will have to collect at least K of the N shares distributed. Then the user can recover the original key. However, using the enhanced Shamir's algorithm, the original cannot be recovered with the main piece K_n . This adds extra security and only the user who has access to this key can recover the decryption key. The authors maintain that the proposed scheme is in accordance with NIST standards in security mechanism, key management system, and key management system survivability.

Analysis and performance evaluation are performed using OpenStack. The private cloud was set up on three Linux machines, one for consumer and the other two as object store and compute servers. Datastores were created in form of database and deployed in the proposed protocol. The authors claim that it is secure and no user except the owner of the key could access the full key by collecting K shares of partial

keys. Even if a component was missing, the owner could regenerate the key. One of the main features is that it limits the amount information accessible to the attacker, which is beneficial to provide more security.

The protocol presented is secure and uses an enhanced version of an existing algorithm. An attacker cannot recover data without the main part of the key. Hence, the security is enhanced. The computation is done on the fly, enabling integrity. However, no results are presented. Future work could be using it to perform a comparative analysis with existing algorithms.

2.8 Enhanced RSA Algorithm with Varying Key Sizes for Data Security in Cloud [11]

This paper presents an enhanced version of the existing RSA algorithm. The major problem that the authors have addressed is that as the size of keys increases to increase security, the time for encryption and decryption is also increased. The proposed algorithm provides a low encryption and decryption time by dividing the files into blocks and enhances the strength of RSA algorithm by increasing the key size.

The original RSA algorithm uses two prime numbers for key generation. The proposed enhanced RSA algorithm uses two prime numbers (P and Q) and apart from those, two more prime numbers (PR_1 and PR_2) are chosen. Four prime numbers are multiplied to calculate $N_1 = P * Q * PR_1 * PR_2$ while $N_2 = P * Q$. A function F is then computed such that $F = (P - 1) * (Q - 1) * (PR_1 - 1) * (PR_2 - 1)$. The public key, E , is then computed such that the greatest common divisor of E and F is 1. The private key, D , is then computed as $D * E = 1 \text{ mod } (F)$. Thus, the public key is (E, N_1) and the private key is (D, N_2) . Encryption is done as $C = ME \text{ mod } (N_1)$ where C is ciphertext and M is the plaintext message. Decryption is done as $M = CD \text{ mod } (N_2)$.

Instead of using random numbers, prime numbers are used as it is more difficult to identify a prime number than a random number. The block size of each file is determined by the key size and is given by $\text{Block Size} = (2 * \text{Key Size}) - 1$. Analyses and results of this enhanced RSA algorithm are presented. It was performed on file sizes varying from 200 to 2000 bit with key sizes varying from 128 to 1024 bits in Java environment. As a comparison with existing methods, ERSA outperforms them. Both the encryption and decryption time are less.

The algorithm presented in this paper enhances an already difficult-to-break encryption algorithm. The proposed algorithm adds extra security making it difficult for attackers to break the key encryption. The results also prove that this algorithm performs well. The paper mentions that it is for files, but it can also be extended to use for key management in the cloud.

2.9 *Secure Management of Key Distribution in Cloud Scenarios [10]*

This paper deals with the issue of key-generated tags in the cloud environment. The problem is that when a T1 tag is generated from a key, another T2 tag is generated from T1, if T1 is lost or damaged we cannot get the first key using just T2. The tag is used to reduce the number of keys held by the user by storing tags on the server. By using tags and a few key numbers, the user can count all the authorized keys in the multi-owner area. As the keys increase the tags and increase the load creation on the user side, two algorithms are introduced for this purpose.

The secure key distribution (SKD) scheme is proposed to strengthen security such as that the user needs to handle one key and one tag. By simply using this information, the user can calculate all the authorization keys without using any information stored on the server. The scheme follows the structure of the tree to calculate the tags. The main idea is to directly calculate the tags reducing the number of tags in the programs. The first algorithm allows the user to calculate children's codes from the parent node. When this is done, add the corresponding tags to the parent node. Tag the leaf node with the parent node.

The second algorithm introduced is for secure key sharing. The main idea of this algorithm is to reduce the number of tags held by the user. This algorithm first calls the tag computation algorithm to count tags and find new parent locations. If the parent node does not have a parent node it means that all tags in this method are computer-based. After that, each parent node is treated as a leaf node and tested by the children's nodes. Finally, the marker is made as a user key and key containing tags. This is still distributed to users where data disassembly can be done.

The analysis of the proposed algorithms is presented. The parameter for analysis is the runtime of the algorithm. Key distribution was performed on varying sizes of keys. Compared with an existing method, SKD performed better in terms of runtime. It was quicker and communication burden is less because tags are not stored at server. For query computation, SKD performed better than the existing method.

The paper presents a novel approach and eliminates the excess number of keys a user must store for each key by using tags. By using proposed method, one key and tag is enough for the user to decrypt data and saves storage costs and burden. The results presented show that the algorithm performs well and is efficient and can be employed by cloud providers. The algorithms presented could be explained better.

2.10 *A Secure Cloud Computing Authentication Using Cryptography [12]*

The authors present a new cryptography algorithm by using elliptic curve cryptography. They aim to improve the existing algorithms by providing a better security

model with minimal computational power for cloud computing. Elliptic curve cryptography uses the algebraic principle of elliptic curve, which includes some variables and coefficients under a finite field. It is a secure public key cryptosystem.

The proposed scheme works as follows:

- Each user is a point on the elliptic curve. Hence the points are generated. As an example, two users A and B are presented.
- The public and private key pair is generated for each user using the points on the elliptic curve. One point is considered as the base point of the user.
- To generate a signature of a message, a cryptographic hash function is used. By selecting a random integer from a given set in the curve, a signature S is computed. This is sent to user B in the cloud for authentication purposes.
- Now that the signature is generated, the next plaintext must be encrypted. The plaintext is encrypted by computing the ASCII value of the text and a random point on the elliptic curve. This is encrypted with the public key of the user using the base point.
- To decrypt the data, the user uses his private key. He calculates the product of the first point in the ciphertext and subtracts it from the second point in the ciphertext. This gives the user plain text.
- Signature verification can also be done by a user. Each user has a signature generated and any other user can verify to provide authentication.

The results are presented in comparison to two asymmetric algorithms RSA and ECC and one symmetric algorithm AES. Analysis was performed by varying key sizes, and it was found that the proposed scheme provides less complexity and computational overhead. For an RSA cryptosystem with 3072 bit key and AES with 128 bit key, ECC provides the same security with 256 bit key. Hence, memory is optimized, and complexity is less. This is beneficial for smaller devices with low complexity.

The algorithm presented uses elliptic curve cryptography, which is documented to be secure. The presented results only take key size into consideration. A better comparison would have been with encryption and decryption time. The aim of reducing complexity is achieved but time is also an important factor, which must be considered.

2.11 *CloudJS Proposed Algorithm for Cloud Encryption Platform [13]*

The Jumbling salting algorithm was founded in 2014 [14, 15], which was initially used for the encryption of password. The aim of creating this algorithm was to secure passwords from dictionary and brute force attacks. The initial version of the algorithm was found to be time-consuming as the encryption and decryption time of the algorithm was high as compared with traditional and popular algorithms like AES

and DES. Later on, this algorithm was used in files [16], credit card authentication [17], large text files [18], images [19]. Recently, the algorithm was also used in cloud environment. The conceptual architecture of Jumbling-Salting algorithm is presented in Ref. [13]. The algorithm is purely based upon the randomization of the characters based on the random value, which is generated from predefined mathematical function. The salting process includes the addition of unique server's timestamp value, which is generated from server's clock. The standalone version of the algorithm is evaluated against the parameters like encryption time, decryption time, throughput, size of ciphertext generated, etc. The cloud version of the algorithm is implemented using virtual machine containers and will be evaluated using cloud-specific parameters like scalability, heterogeneity, etc.

3 Parameters to Evaluate Cloud Encryption Algorithms and Key Management

From the existing study, it is also important to know what are the parameters to evaluate the encryption algorithms with respect to cloud encryption platform. The post part of this study focuses on two aspects of parameters for encryption algorithm evaluation, which was used in the existing cloud encryption frameworks (Fig. 1).

The description of all the parameters is given below.

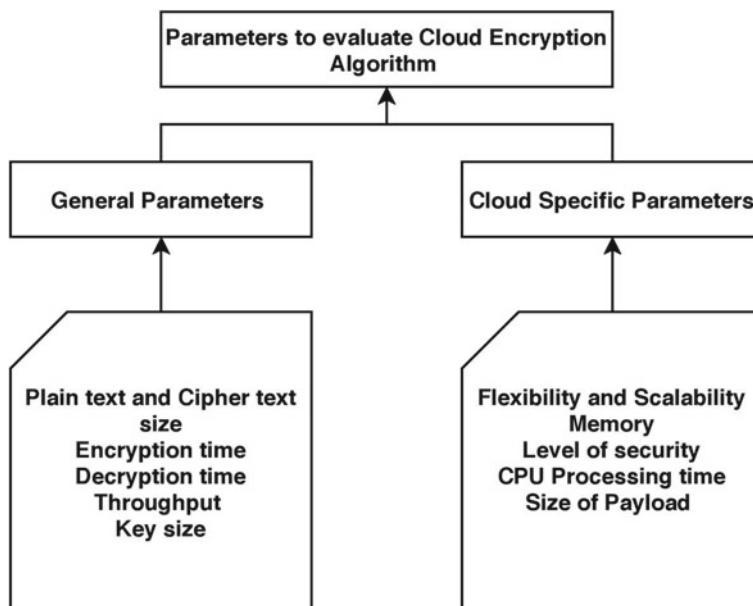


Fig. 1 Categorization of the parameters for evaluating cloud encryption framework

3.1 General Parameters [20–22]

- Plain text and ciphertext size: A good algorithm must have good ratio of ciphertext (encrypted string/text) to plain text since it is susceptible to brute force attack.
- Encryption time: It is the amount of time taken to perform the encryption. It is desirable to have less encryption time for algorithm.
- Decryption time: It is the amount of time to decrypt the encrypted string/text. A less decryption time on server side is desirable.
- Throughput: The encryption time can be used to calculate the encryption throughput of the algorithm. The decryption computation time is the time taken by the algorithms to produce the plain text from the ciphertext. The decryption time can be used to calculate the decryption throughput of the algorithms. Encryption time is used to calculate the throughput of an encryption scheme. It indicates the speed of encryption.
- Key size: It is the bit size value of key that is being used in the encryption algorithm. The ideal key size must be less but it must be random.

3.2 Cloud-Specific Parameters [23, 24]

- Flexibility and Scalability: This is the measure of whether an encryption algorithm can be modified or no mostly with respect to key size.
- Memory: This is the amount of running time memory used by server's CPU in KB.
- Security: The amount of security level implemented in the encryption. Security parameter is majorly depending upon the key size.
- CPU processing time: It is the amount of load time of CPU of cloud server during the encryption. It is desirable to have less CPU processing time for good encryption algorithm.
- Size of payload: This is the sample file size used for encryption for experimental analysis.

4 Classification of Papers

The papers in the survey were very similar. First, we will look at the papers that proposed new algorithms. Papers [9, 11, 12] propose algorithms with enhancements to the existing algorithms. They add a new layer of security without losing the original advantages of the algorithm. The comparisons provided are also helpful in terms of key size, encryption, and decryption time. These are not proposed specifically for key management but can be employed for that purpose.

References [2–4] provide a good insight into the existing mechanisms and services for key management. They clearly and concisely explain the structure and overview

of the existing methodologies. The papers also explain the limitations of current methods and potential solutions for the problems.

Papers [1, 6, 8–10] address the framework of key management and provide some potential solutions for the key management problem. Reference [8] works on all aspects including symmetric and asymmetric keys, individual and group key management and all policies and securities. References [1, 6] address key management for groups of users and propose different approaches to the problem using hypergraphs [1] and dynamic managers [6] while Ref. [4] proposes the use of metadata to keep track of all keys, group or individual. The use of tags and reducing burden on user side is presented in [10]. Paper [9] works on security when keys are stored on cloud side. The work and results proposed show that the schemes are feasible and have to potential to overcome some threats to the cloud computing paradigm.

5 Conclusion

Key management is a very important aspect since it is the core part of cryptography. If any part of the key is exposed to an attacker, data can be leaked or lost. This survey tells us that secure key management is difficult to achieve, and existing methods have some flaws. The proposed methods provide potential solutions to the existing methodologies wherein some of the flaws can be patched. The results presented in the papers surveyed show some promise to patch the vulnerabilities described in the problem description. However, key management is essential for keeping data encrypted on cloud servers and we must provide a high level of security to the customers. Future work should be concentrating on low computation time and complexity combined with a high level of encryption. Such a solution would mean security will be at the highest level and no one can break it. Security can never be absolute; hence, the best we can do is to provide the highest level of security. Any security problem is an open problem and key management is no different.

References

1. Y. Wang, Z. Li, Y. Sun, Cloud computing key management mechanism for cloud storage, in *Third International Conference on Cyberspace Technology (CCT 2015)*, Beijing (2015), pp. 1–4
2. X. Huang, R. Chen, A survey of key management service in cloud, in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China (2018), pp. 916–919
3. A.R. Buchade, R. Ingle, Key management for cloud data storage: methods and comparisons, in *2014 Fourth International Conference on Advanced Computing & Communication Technologies* (IEEE, 2014)
4. S.A. Oli, L. Arockiam, A framework for key management for data confidentiality in cloud environment, in *2015 International Conference on Computer Communication and Informatics (ICCCI)* (IEEE, 2015)

5. O.A. Alzubi, J.A. Alzubi, O. Dorgham, M. Alsayyed, Cryptosystem design based on Hermitian curves for IoT security. *J. Supercomput.* 1–24 (2020)
6. P. Varalakshmi, A.R. Shajina, T. Kanimozhi, A framework for secure cryptographic key management systems, in *2014 Sixth International Conference on Advanced Computing (ICoAC)* (IEEE, 2014)
7. J.A. Alzubi, R. Manikandan, O.A. Alzubi, I. Qiqieh, R. Rahim, D. Gupta, A. Khanna, Hashed Needham Schröder industrial IoT based cost optimized deep secured data transmission in cloud. *Measurement* **150**, 107077 (2020)
8. S. Lei, D. Zishan, G. Jindi, Research on key management infrastructure in cloud computing environment, in *2010 Ninth International Conference on Grid and Cloud Computing* (IEEE, 2010)
9. F. Fakhar, M.A. Shibli, Management of symmetric cryptographic keys in cloud based environment, in *2013 15th International Conference on Advanced Communications Technology (ICACT)* (IEEE, 2013)
10. Z. Cui, H. Zhu, J. Yu, Secure management of key distribution in cloud scenarios, in *Proceedings of 2014 International Conference on Cloud Computing and Internet of Things* (IEEE, 2014)
11. I.G. Amalarethinam, H.M. Leena, Enhanced RSA algorithm with varying key sizes for data security in cloud, in *2017 World Congress on Computing and Communication Technologies (WCCCT)* (IEEE, 2017)
12. M. Chakraborty, B. Jana, T. Mandal, A secure cloud computing authentication using cryptography, in *2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR)* (IEEE, 2018)
13. R. Gupta, A. Doshi, S. Dharadhar, P. Churi, Conceptual architecture of cloud JS encryption algorithm. *Int. J. Adv. Sci. Technol.* **29**(3), 7625–7640 (2020)
14. P. Churi, M. Kalelkar, B. Save, JSH algorithm: a password encryption technique using jumbling-salting-hashing. *Int. J. Comput. Appl.* **92**(2) (2014)
15. P.P. Churi, V. Ghate, K. Ghag, Jumbling-salting: an improvised approach for password encryption, in *2015 International Conference on Science and Technology (TICST)* (IEEE, 2015), pp. 236–242
16. M.U. Bali, M.N. Udgata, M.P.P. Churi, Symmetric jumbling-salting encryption algorithm for files, in *2018 Fifth HCT Information Technology Trends (ITT)* (IEEE, 2018), pp. 82–86
17. M.N. Prasad, M.R. Oruganti, M.S. Shah, M.Y. Pavri, P. Churi, Improvised e-commerce transaction security using JS secure algorithm, in *2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)* (IEEE, 2018), pp. 1–7
18. P.P. Churi, *Performance Analysis of Data Encryption Algorithm*
19. M. Vartak, R. Rishabh, C. Prathamesh, Image encryption using jumbling-salting algorithm. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)*
20. A.R. Wani, Q.P. Rana, N. Pandey, Cloud security architecture based on user authentication and symmetric key cryptographic techniques, in *2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)* (IEEE, 2017), pp. 529–534
21. S. Belguith, A. Jemai, R. Attia, Enhancing data security in cloud computing using a lightweight cryptographic algorithm, in *The Eleventh International Conference on Autonomic and Systems* (2015), pp. 98–103
22. L. Zhou, V. Varadharajan, M. Hitchens, Achieving secure role-based access control on encrypted data in cloud storage. *IEEE Trans. Inf. Forensics Secur.* **8**(12), 1947–1960 (2013)
23. A. Nandgaonkar, P. Kulkarni, *Encryption Algorithm for Cloud Computing*
24. A. Altigani, S. Hasan, S.M. Shamsuddin, B. Barry, A multi-shape hybrid symmetric encryption algorithm to thwart attacks based on the knowledge of the used cryptographic suite. *J. Inf. Secur. Appl.* **46**, 210–221 (2019)

Domain-Controlled Title Generation with Human Evaluation



Abdul Waheed , Muskan Goyal , Nimisha Mittal , and Deepak Gupta

Abstract We study automatic title generation and present a method for generating domain-controlled titles for scientific articles. A good title allows you to get the attention that your research deserves. A title can be interpreted as a high-compression description of a document containing information on the implemented process. For domain-controlled titles, we used the pre-trained text-to-text transformer model and the additional token technique. Title tokens are sampled from a local distribution (which is a subset of global vocabulary) of the domain-specific vocabulary and not global vocabulary, thereby generating a catchy title and closely linking it to its corresponding abstract. Generated titles looked realistic, convincing, and very close to the ground truth. We have performed automated evaluation using ROUGE metric and human evaluation using five parameters to make a comparison between human and machine-generated titles. The titles produced were considered acceptable with higher metric ratings in contrast to the original titles. Thus we concluded that our research proposes a promising method for domain-controlled title generation.

Keywords Automatic title generation · Natural language generation · Transformer · Summarization technique · T5 model · Domain control · Additional token technique · Human evaluation

1 Introduction

The analysis of literature is a crucial practice for researchers to determine relevant publications for the research topic. The title of the research papers becomes really significant due to the availability of scientific papers in abundance. Researchers can automatically determine the importance of a paper by its title rather than reading the

A. Waheed · M. Goyal · N. Mittal · D. Gupta

Computer Science Department, Maharaja Agrasen Institute of Technology, Delhi, India
e-mail: e.abdul@protonmail.com

D. Gupta
e-mail: deepakgupta@mait.ac.in

whole document [1–3]. The accuracy of the title influences the number of potential readers and therefore the number of citations [1, 4]. That’s why it’s important for the researchers to produce a good title, however, people spend very little time on it [3]. This leads to non-informative headings that do not take the entire content of the scientific paper into account.

To establish a title for a scientific paper, one has to understand the objective and characteristics of the paper, so that the meaning of the paper can be distilled into the title with only a few words. A suitable title for a particular paper must convey the core meaning of the paper briefly. The automatic title generation task has traditionally been closely linked to the conventional summarization technique because it can be considered as scientific paper compression to represent its content [5–7]. For a given paper, a short and concise description, conveying the complete details of the text in just a few words, must be made. The task cannot, therefore, be regarded as simple.

It is crucial to identify the type of information/domain of each text unit to generate a suitable title. The performance of automatic title generation can become unsatisfactory due to the differences between the domains of scientific papers. Such differences include diverse vocabulary, different forms of grammar, and various ways of expressing identical concepts. Therefore, it is important to use domain-specific vocabulary to generate the title for scientific papers. Although automated title creation is not capable of replacing an author’s skills in creating a title, it is helpful to propose a title.

In this research, we propose to utilize the text-to-text transformer model (T5 model) for domain-controlled title generation. The text-to-text transformer model converts all the NLP tasks to a single text-text format where text is used as input and output. This formatting allows the T5 model to perform a number of tasks into its framework, including the summarization task. For domain control, we used an additional token method which is discussed in Sect. 3.3. Since an abstract of a scientific paper describes the author’s work and presents all key arguments and relevant findings [8], we used it to generate a suitable domain-controlled title for papers. Several researchers have worked on automatic title generation. We examine some of these works below.

1.1 Related Work

Liqun and Wang proposed a method called DTATG in [9] to produce titles. DTATG was an unsupervised method that produced syntactically correct titles very easily. This method used sentence segmentation to draw a limited number of central sentences that express the core meanings of the text. DTATG created a dependency tree for each of these sentences and extracted several branches for trimming purposes with a dependency tree compression model. The authors also designed a title test to figure out if the trimmed sentence can be used as a title. The title with the top-ranked score was selected as the final title. DTATG created titles similar to human-generated titles. However, DTATG was constrained to the use of central phrases.

In [10], Putra et al. proposed a method for the creation and classification of rhetorical corpus structures. The experiment was directly integrated into the role of automatic title generation for scientific papers. Each sentence was categorized into one of the three groups: OWN_MTHD (method), AIM (purpose), and NR (not relevant). Firstly the abstract of the scientific papers was annotated with the corresponding category. Then features like rhetorical patterns and formulaic lexicon statistics were decided and analyzed for classification purposes. Finally, several supervised learning algorithms were utilized to build the classification model. The models were tested using tenfold cross-validation and resulted in a weighted average F-measure between 0.70 and 0.79.

Liu et al. [11] proposed a sentence2vec-enhanced Quality-Diversity Automated Summarization (QDAS) model and attempted to implement transfer learning for the Wikipedia title Generation task. Summaries from paragraphs were derived by extractive summarization method and sequence labeling data was provided to the model for title generation. The system involved only general pre-processing, such as sentence splitting and word segmentation, and could be implemented in a multilingual environment. The authors fine-tuned the BERT-based CRF model for Wikipedia article title generation.

Chen et al. [12] presented a method for domain adaptation using artificial titles for title generation. The authors discussed the strategies for modifying the encoder-decoder model for text generation from a domain marked source to an unlabeled domain target. Sequential training was utilized in order to capture the unlabeled target domain's grammatical form. For the title generation task, an encoder-decoder RNN model with domain control was used. The encoder collected data from the source and the decoder learned to construct summary captions. The source data and unlabeled target domain information were encoded as their definition representations by a bidirectional LSTM, and the domain classifier attempted to learn to distinguish between the representations of two domains.

In [13], Gehrmann, Sebastian et al. proposed a method to generate titles for short sections of long documents. The authors aimed at designing techniques in a low-resource system for section title generation. They first picked the most influential sentence and then performed a removal-based compression on an extractive pipeline. The Semi-Markov Conditional Random Field compression method was utilized. The method depended upon unsupervised textual representations such as ELMo or BERT to eliminate the need for the design of the complex encoder-decoder.

Most of the above works are centered on how relevant terms can be extracted in the article for the title output without recognizing the domain of the text. The domain offers a category of material that reflects the communication purpose conveyed by a paper to the reader. It is alleged that the sentence domain in the document could boost the effectiveness of automatic title generation by including a certain sort of information in the document title. In this research, we consider domains in the form of information types communicated by abstracts of scientific papers. Also, we used a transformer-based architecture model (T5 model) instead of using sequence-to-sequence models (RNN, LSTM) to perform automatic domain-controlled title generation. RNN/CNN handle sequences word-by-word sequentially which is an

obstacle to parallelize. Transformers remove recurrence and convolution entirely and substitute them with self-attention to assess the dependencies between inputs and outputs. Transformer achieve parallelization by replacing recurrence with attention and encoding the symbol position in sequence. This in turn leads to significantly shorter training time and better performance.

1.2 Contributions

While considering the above works, we have proposed a domain-controlled automatic title generation method. The key contributions to this paper are mentioned as follows:

1. Propose a method for automatic domain-controlled title generation using transformers.
2. Provide meta-information about the domain using the additional token technique.
3. Compare the titles generated by the model with human-generated titles by performing the human evaluation.

The remaining exhibition is the following. The background is presented in Sect. 2. The proposed model is discussed in Sect. 3. Section 4 deals with the simulation model, followed by Sect. 5 with results and discussion. Section 6 addresses the conclusions and possible future directions.

2 Background

Natural Language Processing (NLP) is a facet of Artificial Intelligence and is responsible for computers and machines to understand, interpret, and further incorporate human languages. NLP is based on different disciplines, like computer science and machine-translated linguistics. It further attempts to narrow the communication gap between computers and humans.

2.1 Attention

Attention [14] proposed in machine translation alleviates the bottleneck of seq2seq learning. It allows us to look at hidden states of the source sequence and they are parsed as an additional input in the form of weighted average to the decoder. It is not restricted to input sequence and the concept of self-attention can be incorporated to analyze surrounding words to obtain contextually sensitive word representations. Transformer architecture [15] consists of various self-attention layers.

2.2 *Pre-trained Language Models*

The pre-trained word embeddings initialize only the first layer of the model and the embeddings are generally context-agnostic. Pre-training always learns the important parameters of deep neural networks and is further fine-tuned for various tasks. They were proposed in 2015 [16] but their benefits across various tasks were recently observed. Pre-training involves improved model initialization that boosts generalization performance and convergence speed for the tasks. Being language models, they can solely rely on unlabeled corpora, hence they are beneficial when labeled data is scarce.

2.3 *Transformer Architecture*

The Transformer [15] is a model architecture that solves the problem of parallelization by integrating self-attention techniques with convolutional neural networks. Before the Transformer was introduced, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) were used for sequence transduction. Similar to RNNs and CNNs, there is an encoder and decoder in the transformer where each block includes a multi-head attention block, skip connection, and normalization layer followed by a feed-forward block. With the help of multi-head attention and positional embedding, the transformer network has the capability of processing the data parallelly. Often, transformer-based models contain stacked encoders and decoders.

The self-attention mechanism in this model is responsible for correlating each input token regardless of their position. Initially, all the words are represented using high-dimensional vectors or embeddings. This process is performed in the bottom-most encoder that is subsequently passed to all the succeeding encoders which perform self-attention to generate new representation considering contextual information. The decoder having similar architecture operates likewise but generates one word at a time by attending the encoded contextualized representation and previously generated tokens.

3 Proposed Method

This section presents the methodology and architecture used in the research. A detailed overview of the model and the process used to make the overall concept comprehensible has been given. The present study discusses title generation as a summarization process. Our suggested architecture comprises three modules integrating a primary pipeline for summarization: pre-processing, training of the T5 model, and generation of the title (summary). The flowchart of the proposed method is shown in Fig. 1. Each module is defined in detail in the following sections.

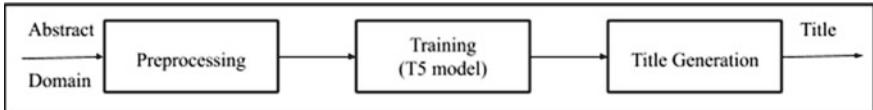


Fig. 1 Flowchart of title generation using T5. Pre-processing involves mainly tokenization which is based on subword tokenization

3.1 Pre-processing: Subword Tokenization

The tokenization of subwords is a strategy that embraces uncertainty in data. Subword tokenization breaks the text into subwords. For instance, terms like *smartest* are segmented as *smart-est*, *lower* as *low-er*, etc. Transformer-based models depend on subword tokenization algorithms for vocabulary generation. This method divides OOV (out of vocabulary) terms into subwords and depicts the term in these subwords. The length of input and output phrases is less than the tokenization of the character after subword tokenization. This method uses the most frequently occurring character or sequence iteratively.

3.2 T5 Model

T5: Text-to-text-Transfer-Transfer model was proposed by Raffel et al. [17] to reframe all NLP-based activities to a single text-to-text format where text strings are both input and output. This formatting enables the T5 model to cast various tasks into this framework including machine translation, classification task, regression task, and summarization task. This framework uses the same model, hyperparameters, and loss function on the tasks mentioned.

The model architecture is similar to standard vanilla encoder-decoder transformers with similar computational cost and considering half parameters. The input sequence is initially mapped to the embedding sequence and transferred to the encoder afterward. The encoder comprises a stack of blocks, and each block consists of two subcomponents: a self-attention layer and a feed-forward network. Normalization of the layer is introduced to each subcomponent's input. Normalization only includes rescaling of the activations and no additive bias is introduced. This is followed by a residual skip connection that adds the subcomponent's input to its output. This model considers a different position embedding scheme and these are passed as parameters across all layers of the model.

The structure of the decoder is similar to the encoder except it incorporates the standard attention process after the self-attention layer encoder's output. The self-attention in the decoder is a type of auto-regressive or causal self-attention that only enables the algorithm to attend to previous outputs. The final decoder block output is fed into a dense layer with a softmax output, the weights of which are shared with the

input embedding matrix. Hence, the T5 model is similar to the original Transformer with an exception of the removal of Layer Norm bias and using different position embedding methods.

3.3 Additional Token Technique for Domain Control

Domain control or often called adaptation is an important research area in the processing of natural languages. The arXiv dataset has a category/domain label corresponding to each paper. By controlling the domain, title tokens will be sampled from the local distribution of the domain-specific vocabulary (which will be a subset of global vocabulary) rather than global vocabulary. The produced title will therefore be attractive and realistic, closely associated with its corresponding abstract.

Now, it would be cumbersome and time-consuming to train one separate model for each type of domain. The solution to this problem is to add a reference to the desired domain to the input text. The reference acts as a control tag specific to each type of domain. This helps the model to learn the relation between the control tags and the following text and only a preferred control tag must be defined by the user to specify the type of text that is to be generated. This control tag is the meta-information about the domain that we provide to the model. The tags are supplied using the additional token method technique.

The additional token method [18] requires the addition of an artificial token at the end of each abstract which enables the model to pay attention to the domain of each title and abstract. The model reads the abstract with the associated domain tag @DOMAIN-NAME. Domain tags are already defined in the ArXiv metadata file, corresponding to each paper. Although simple, this approach has proved to be successful [19, 20]. Note that the T5 model does not include any information about the domain. It includes domain information through this additional token approach. Now, we need to maximize the probability of title using abstract and domain.

Domain Controlled,

$$\text{loss} = -\log \sum_{t=1}^{|y|} P(y_t^* | y_{<t}^*, x, d) \quad (1)$$

Without domain control,

$$\text{loss} = -\log \sum_{t=1}^{|y|} P(y_t^* | y_{<t}^*, x) \quad (2)$$

where

$y_t^* = y^{<1>} , y^{<2>} , \dots , y^{<n>}$ denotes tokens in generated title
 $x = x^{<1>} , x^{<2>} , \dots , x^{<m>}$ denotes tokens of the abstract

d denotes domains (examples astro-ph: Astrophysics, cs.CC: Computational Complexity, cs.CL: Computation and Language).

3.4 Training Procedure

Aforementioned, a T5 model takes input and produces output in text format. A unique prefix tag is used to define and train the model on different tasks. The format of input text passed to the model consists of three parameters, prefix, input text, and target text. This makes training of the model simple as the model only requires a change in the prefix tag to run a specific task. The format of input can be stated as <prefix>:<input_text>.

The first step is to feed the text input into a layer of word embedding that generates a vector representation of each word. Further, positional encoding is used to inject positional data into the input embeddings. The encoder layer present transforms input sequences to abstract continuous representation. The decoder is responsible for producing text sequences that are capped off with a linear and a softmax layer. Moreover, the decoder consists of a start token, a list of previous outputs that play the role of inputs along with encoder outputs containing attention information. It stops decoding when a token is produced as an output.

In our implementation, we used Beam Search with beam size 4 as the decoding method because it eliminates the likelihood to miss hidden word sequences with high probability. For instance, consider the abstract and title pair illustrated below. This abstract is further passed to the T5 model after appending the appropriate domain tag as represented to predict domain-controlled title of the scientific paper. The training loss graph of the model for domain control and without domain control is shown in Fig. 2.

Abstract:

This paper proves that labeled flows are expressive enough to contain all process algebras which are a standard model for concurrency. More precisely, we construct the space of execution paths and of higher dimensional homotopies between them for every process name of every process algebra with any synchronization algebra using a notion of labeled flow. This interpretation of process algebra satisfies the paradigm of higher dimensional automata (HDA): one non-degenerate full n -dimensional cube (no more no less) in the underlying space of the time flow corresponding to the concurrent execution of n actions. This result will enable us in future papers to develop a homotopical approach to process algebras. Indeed, several homological constructions related to the causal structure of time flow are possible only in the framework of flows.



Fig. 2 Training loss

Actual Title:

Toward a homotopy theory of process algebra

The T5 model reads the Abstract appended with the appropriate domain tag (@DOMAIN-NAME) of the scientific paper.

Input (Abstract):

This paper proves that labeled flows are expressive enough to contain all process algebras which are a standard model for concurrency. More precisely, we construct the space of execution paths and of higher dimensional homotopies between them for every process name of every process algebra with any synchronization algebra using a notion of labeled flow. This interpretation of process algebra satisfies the paradigm of higher dimensional automata (HDA): one non-degenerate full n -dimensional cube (no more no less) in the underlying space of the time flow corresponding to the concurrent execution of n actions. This result will enable us in future papers to develop a homotopical approach to process algebras. Indeed, several homological constructions related to the causal structure of time flow are possible only in the framework of flows. @domain: math.AT math.CT.

Output (Predicted Title):

Labeled flows and homotopical approach to synchronization algebras.

4 Simulation Model

4.1 *Simulation Setup*

The entire model is trained and tested on an Nvidia Tesla P100 GPU with 16 GB GPU VRAM using python language. The overall parameters of the model and dataset are defined below.

4.2 *Parameters*

The T5 model used various parameters in the training and testing process. By adjusting these parameters, the accuracy of the model can be determined. The maximum sequence length of the input (abstract) is set to 512 and the maximum sequence length of output (title) is set at 20. For both training and testing, a batch size of 8 is used. The model is trained with a learning rate of 1e-5 for a total of 5 epochs. An evaluation occurred once for every 1000 training steps.

4.3 *Dataset*

The model is trained on the popular ArXiv collection [21]. ArXiv has served the general and research community with free access to scholarly publications from the major fields of physics to several computer sciences disciplines and all that goes between them, including electrical engineering, computational biology, mathematics, and statistics. This large corpus of data provides substantial, but often daunting, depth. This dataset contains the original data from ArXiv. Since the complete dataset is very huge (with 1.7 million articles), it provides a JSON-format metadata file. For each paper, this file contains an entry with key features such as names of articles, authors, domains/categories, abstracts, complete text PDFs, etc. For this research, we used 1,10,000 samples from the dataset that is further divided into 90% for training and 10% for the purpose of validation.

5 Results and Discussion

5.1 *Automated Evaluation*

Automatic evaluation is performed for evaluating systems when the output is the text that is generally referred to as a sequence to sequence or string transduction problem. For evaluating the results, generated text from the model is compared with

Table 1 Results of the automatic evaluation. ROUGE score with abstract + domain is significantly better than without a domain

Model	Data	ROUGE-1	ROUGE-L
T5	Abstract	28.5	16.6
	Abstract + Domain	31.5	21.6

target (source) or reference text. There are various evaluation metrics, but this paper focuses on the ROUGE metric.

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [22] is an intrinsic metric that is recall focused. This metric measures the number of n-grams or words present in the target text summaries that appeared in the generated text by machines. There are three prominent types of Rouge scores including Rouge-N, Rouge-L, and Rouge-S. The N-gram overlap between the machine summary and the target summary is represented in Rouge-N. Rouge-N contains Rouge-1 and Rouge-2, which is generated and goal text correspond to the contrast of unigrams and bigrams, respectively. The longest matching sequence of terms using LCS is measured by Rouge-L. Rouge-S focuses on skip-bigrams based co-occurrence statistics. This paper uses only Rouge-1 and Rouge-L metrics for comparison. The scores of the ROUGE metric can be seen in Table 1.

We trained the T5 model for title generation in two different ways. In the first one, we generated the title directly from abstract text where we observed that the tokens in the generated title were not specific to the domain of the paper. Hence, to address this issue we controlled the domain using additional token technique and with this approach we noticed significant improvement. It can be inferred from Table 1 that ROUGE-1 and ROUGE-L scores obtained for domain-controlled abstracts (Rouge-1: 31.5 and Rouge-L: 21.6) are considerably higher to the domain-less abstracts.

5.2 Human Evaluation

The interest in analyzing NLG (Natural Language Generation) texts in recent years has grown by contrasting them to a corpus of human texts [23]. As with other NLP fields, the benefits of automated corpus-based assessment are that it is theoretically both cheaper and faster than human-based assessment and is also repeatable. Corpus tests have been first used in the NLG, where the texts scanned from a corpus have fed the output of the parser into the NLG system and have then been compared to the original text of the corpus. In the NLG group, such corpus-based evaluations were often criticized. Reasons for criticism include the fact that the regeneration of a parsed text is not a true NLG task; that texts can vary considerably from corpus text but are still successful in achieving the communication purpose of the method and corpus texts often are not of good quality enough to shape a practical evaluation.

Table 2 Human evaluation scores. Compared with human-scaled, T5 with domain generated titles are scored better than human-generated titles by human evaluators on almost all parameters

Model	Coherence	Relevance	Fluency	Semantic adequacy	Overall quality	Mean score
Human	4.2	4.29	4.11	4.05	4.16	4.16
T5 + Domain	4.53	4.63	4.58	4.43	4.6	4.55
Scaled	1.08	1.08	1.11	1.09	1.11	1.09

Human Evaluation is another approach to evaluate NLG (Natural Language Generation) based systems. This involves performing quality surveys of the generated output using human annotators. In this approach, generated results are presented to the people who assess the quality of the text on different criteria. Using intrinsic and extrinsic approaches [23], human evaluation of natural language generation systems can be performed. Intrinsic methods seek to test the performance properties of the system by asking participants about the fluency of the output of the system in a questionnaire. Extrinsic methods aim to measure the effect of the system by evaluating the degree to which the system accomplishes the overarching task for which it was created. Extrinsic testing is the most time-consuming and cost-intensive of all possible tests; hence it is very rare.

We conducted an intrinsic human evaluation with 40 participants. The academic background and field of study of each participant were also recorded. We prepared a form that paired 2 abstracts with the original and predicted title. The nature of the title (original or predicted) was not known to any of the participants. The participants judged and rated each title from 1 (very bad) to 7 (very good). The judgments were based on various factors like coherence, relevance, fluency, semantic adequacy, and overall quality of the title. The aggregated mean score was first calculated for each factor, and then for each of the original and predicted titles. The values are shown in Table 2.

In the process of human evaluation, each factor has its own significance. Coherence determines whether or not the title is semantically meaningful, and relevance indicates how applicable the title is to the abstract. Two titles may equally effectively express the underlying intention while varying in fluency. Therefore, the titles were judged for their readability with fluency (able to understand the meaning of the title) and semantic adequacy (the title is a sufficient representative of abstract) as its factors. It is often difficult for human annotators to differentiate between the different quality measures. This is why the overall quality of a title is evaluated directly.

The most suitable number of answers can rely on the task itself, but for most tasks, 7-point ratings are the best. Several studies suggest that 7-point ratings optimize reliability and discriminatory power [24–26]. One 7-point likert question has 7 points for discrimination but 4 7-point likert questions have $4 * 7 = 28$ discrimination points. Therefore, one 7-point likert rating was used for each of the 4 titles in human evaluation form.

It is clearly evident from Table 2 that the predicted title has higher values of evaluation metrics (coherence, relevance, fluency, semantic adequacy, and overall quality) in comparison to the original title. Overall, on normalizing the difference between the values of human-generated and machine-generated titles, the scaled value is in the range from 1.08 to 1.11. Moreover, the mean score of the title predicted using the proposed model is greater than human-generated titles. This distinctly depicts that the title produced using the T5 model and additional token is more appropriate and useful for the authors of scientific papers.

6 Conclusion and Future Work

In this research, we proposed the use of the T5 model for automatic domain-controlled title generation for scientific papers. We have used the ArXiv dataset for our research and analysis. Scholars, professional authors, students, and teachers may use this method. Sparseness is the biggest struggle of the title generation. For a document with several optional words, it is important to create a brief and succinct title. An abstract of a scientific paper contains the goal and method of the research, along with all the key findings. Therefore, we utilized a scientific paper's abstract to produce the title for the paper.

We also identify the communication purpose of the authors bypassing the domain of the paper as the meta-information. The quality of the automatic title generated improves because the title tokens are sampled from the local vocabulary of the domain, rather than the global vocabulary. The titles generated from the model looked satisfactory and realistic.

To analyze the results we used automated and human evaluation. For automated evaluation, we calculated ROUGE scores to prove that our approach leads to high performance in the generation of automatic domain-controlled titles. In the human-subject analysis, we used 40 participants with diverse reading capabilities and academic backgrounds. We noticed that our generated titles have higher values of evaluation metrics (coherence, relevance, fluency, semantic adequacy, and overall quality) in comparison to the original title.

Although the abstract is a brief representation of the paper, writers may have written the abstract hastily. Therefore, it is suggested to use the complete paper text along with more refined weighing, title generation, and selection methods, in order to better capture the prominent tokens of a document in a particular domain. Another task that can be addressed in the future is to create new techniques that deliver titles which are more close to human-generated titles.

References

1. J.W. Putra, M.L. Khodra, Automatic title generation in scientific articles for authorship assistance: a summarization approach. *J. ICT Res. Appl.* **11**, 253–267 (2017)
2. R. Hamid, M. Jamali, M. Nikzad, Article title type and its relation with the number of downloads and citations. *Scientometrics* **88**, 653–661 (2011)
3. H. Xu, E. Martin, A. Mahidadia, Extractive summarisation based on keyword profile and language model, in *HLT-NAACL* (2015)
4. C. Paiva et al., Articles with short titles describing the results are cited more often. *Clinics* **67**, 509–513 (2012)
5. R. Jin, A. Hauptmann, *Headline Generation Using a Training Corpus*. Carnegie Mellon University (Journal Contribution). <https://doi.org/10.1184/R1/6606059.v1>
6. M. Witbrock, V. Mittal, Ultra-summarization (poster abstract): a statistical approach to generating highly condensed non-extractive summaries, in *SIGIR'99* (1999)
7. J. Kupiec, et al., A trainable document summarizer, in *SIGIR'95* (1995)
8. J.W.G. Putra, K. Fujita, Scientific paper title validity checker utilizing vector space model and topics model, in *Proceedings of Konferensi Nasional Informatika (KNIF)* (2015), pp. 69–74
9. L. Shao, J. Wang, DTATG: an automatic title generator based on dependency trees, in *KDIR* (2016)
10. J.W.G. Putra, M.L. Khodra, Rhetorical sentence classification for automatic title generation in scientific article. *TELKOMNIKA Telecommun. Comput. Electron. Control* **15**, 656–664 (2017)
11. W. Liu, et al., Multi-lingual wikipedia summarization and title generation on low resource corpus, in *Proceedings of the Multiling 2019 Workshop, Co-located with the RANLP 2019 Conference*, pp. 17–25
12. F.R. Chen, Y.-Y. Chen, Adversarial domain adaptation using artificial titles for abstractive title generation, in *ACL* (2019)
13. S. Gehrmann, et al., Improving human text comprehension through semi-Markov CRF-based neural section title generation (2019), [arXiv:1904.07142](https://arxiv.org/abs/1904.07142)
14. D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in *ICLR* (2015)
15. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, I. Polosukhin, Attention is all you need, in *Advances in Neural Information Processing Systems* (2017)
16. A.M. Dai, Q.V. Le, Semi-supervised sequence learning, in *Advances in Neural Information Processing Systems NIPS'15* (2015)
17. C. Raffel, et al., Exploring the limits of transfer learning with a unified text-to-text transformer (2019), [arXiv:1910.10683](https://arxiv.org/abs/1910.10683)
18. C. Kobus, et al., Domain control for neural machine translation, in *RANLP* (2017)
19. R. Sennrich, et al., Controlling politeness in neural machine translation via side constraints, in *HLT-NAACL* (2016)
20. M. Johnson et al., Google’s multilingual neural machine translation system: enabling zero-shot translation. *Trans. Assoc. Comput. Linguist.* **5**, 339–351 (2017)
21. C.B. Clement, et al., On the use of ArXiv as a dataset (2019), [arXiv:1905.00075](https://arxiv.org/abs/1905.00075)
22. C.-Y. Lin, ROUGE: a package for automatic evaluation of summaries, in *Proceedings of the ACL Workshop: Text Summarization Branches Out* (2004)
23. C. Lee, et al., Best practices for the human evaluation of automatically generated text, in *INLG* (2019)
24. G.A. Miller, The magical number seven plus or minus two: some limits on our capacity for processing information. *Psychol. Rev.* **63**(2), 81–97 (1956)
25. P. Green, V. Rao, Rating scales and information recovery—how many scales and response categories to use? *J. Mark.* **34**, 33–39 (1970)
26. R.W. Lissitz, S. Green, Effect of the number of scale points on reliability: a Monte Carlo approach. *J. Appl. Psychol.* **60**, 10–13 (1975)

A Big Data Query Optimization Framework for Telecom Customer Churn Analysis



Aarti Chugh, Vivek Kumar Sharma, Manjot Kaur Bhatia, and Charu Jain

Abstract One of the most popular social platforms is Twitter and telecom companies are using different algorithms and techniques for doing customer sentiment analysis over big pools of data coming through it. The biggest challenge is tweets are unstructured and queries should run accurately and fast. Our research introduces a new query analytical framework capable to tackle the big data challenges by leveraging deep learning technique, recurrent neural network and metaheuristic approach, spider monkey optimization algorithm. DeepRNN is used during tweet classification as they are capable to work with big data, and spider monkey optimization is applied for training network weights to optimize speed and get accurate results. Experimental results are compared with the existing deep convolutional network model and results show that complex analysis is conducted with good flexibility and performance. The model has optimized the training time of recurrent networks to a minimal value, i.e. 0.3 s.

Keywords Big data · Query optimization · Telecom · Customer churn rate · Deep learning · Twitter · Sentiment analysis · RNN · Spider Monkey optimization

A. Chugh (✉) · C. Jain
Amity University, Gurgaon, Haryana, India
e-mail: achugh@ggn.amity.edu

C. Jain
e-mail: cjain@ggn.amity.edu

V. K. Sharma
Jagannath University, Jaipur, India
e-mail: vivek.sharma@jagannathuniversity.org

M. K. Bhatia
Jagan Institute of Management Studies, Rohini, Delhi, India
e-mail: manjot.bhatia@jimsindia.org

1 Introduction

Churn rate is an important consideration in the telephone and cell phone services industry. Often, telecom companies analyze customer reviews, opinions, favorites, etc., that are collected from social platform—Twitter for making decisions to improve their products/services and retain customers [1]. To process such extremely large data exploding in different formats, not only the right set of data gathering, processing and analysis tools are required but also highly scalable and massively parallel processing systems are required [2].

Telecom sector is already exploiting the power of big data. They are using different algorithms, tools and techniques for querying and doing sentiment analysis over big pools of data coming through Twitter [1, 3]. But, there is no standard model for telecom companies, which addresses the challenges of processing and querying such big data fast and accurately. Several research focus on the design of new twitter sentiment analysis techniques for improved results in terms of accuracy, speed and other parameters [4]. Many of these are based on supervised and unsupervised machine learning, deep learning or the lexicon-based approach [4, 5]. These models can classify the sentiments in the text as positive, negative or neutral. All of these methods have given good results, but very less research exists for big and complex datasets [2].

Our research introduces a big data query optimization approach that optimizes the sentiment analysis process. Query optimization is carried out in different ways in big data infrastructures due to heterogeneous data [6, 7]. Moreover, style of queries is also different from application to application. Companies use Twitter sentiment analysis to predict sentiments and get answer to their queries like number of customers giving positive feedback, which services people discuss mostly, etc. Efficiency, training time, accuracy, query response time, etc. are some parameters in the sentiment analysis, which can be improved during query optimization. Thus, in sentiment analysis process, query optimization can be done in two phases: first during classification phase and second during information retrieval while responding to queries. The proposed algorithm is capable of working with unstructured big data, i.e. dataset prepared by collecting tweets. The model is founded on a hybrid consisting of deep recurrent neural networks [8, 9] and spider monkey optimization (SMO) [10]. Deep recurrent neural networks (DeepRNNs) are employed during classification process because they do not put restriction on input feature vector and possess time-based context to enhance predicted meaning [8, 11]. Furthermore, these proved to work better with sequential inputs, like speech and text [11]. SMO is a metaheuristic approach that aims to competently explore the search space to converge fast to a near-optimal solution [10, 12, 13]. SMO is considered as a flexible algorithm among the swarm intelligence-based algorithms. Its linear nature makes it highly capable for any number of training data with good computing speed [13]. Hence, we employ SMO for training weights in DeepRNN during classification phase. The matching phase takes test query and finds a match from trained dataset.

This research has the following main contributions.

- A new approach was proposed by hybridizing DeepRNN and SMO.
- The query optimization hybrid model applied for optimization of sentiment classification process of big data comprising of customer tweets.

The rest of this paper is organized as follows. Section 2 discusses related work on this topic. Our proposed hybrid query optimization methodology is covered in Sects. 3 and 4. The process of the experiment and result analysis is shown in Sect. 5. Finally, Sect. 6 concludes the paper.

2 Related Work

Jianqiang et al. [14] used DCNN over five Twitter datasets for generating word embedding based on unsupervised learning, where relative semantic relations were found among different words from tweets. The feature set comprised of word embeddings, word sentiment polarity and n-gram features. Ahmad et al. [15] prepared the model using Spark environment for 70 TB dataset provided by Syria Tel Telecom Company. Different network-based features and features related to IMEI data were extracted. Authors have investigated a number of algorithms such as Decision Tree, Random Forest, Gradient Boost Machine Tree and XGBoost tree. XGBOOST tree model achieved the best results in all measurements giving AUC value as 93.301%. Vidya et al. [16] worked on sentiment analysis of Twitter data for measuring brand reputation of mobile providers. Modeling and classification for five important services were done using three classifiers based on Naïve Bayes, SVM and decision tree, and SVM had given a better performance. Virmani et al. [17] propounded a query processing model for unstructured data using a combination of natural language processing, machine learning techniques, ranking algorithm and Latent Dirichlet Algorithm. The model takes the user queries through query processing system interface, passes it to content-based semantic matchmaker phase, then applies machine learning classifier and ranking phase provides desired information. Sasikala et al. [18] recommended sentiment analysis system for product review in three arrangements named grade-based, content-based and collaboration-based. The authors used two methodologies—deep learning modified neural network (DLMNN) during classification phase to improve accuracy and improved adaptive neuro-fuzzy inferences system (IANFIS) to help in predicting future for online products. Spider monkey optimization algorithm was applied during feature extraction phase. The system suffers from a drawback of not identifying complete sentiments from given text. Misha and Verma [19] applied a data optimization approach for sentiment analysis. The optimization was done through genetic algorithm during feature selection in which undesirable characteristics were removed from extracted Twitter feature set. Rao et al. [20] devised a combination of recurrent neural network and the spider monkey-bird swarm algorithm for heartbeat classification based on cardiac arrhythmia and achieved 95% accuracy. Several nature-inspired algorithms are useful for various

engineering solutions like [21] uses cat swarm and grey wolf optimization algorithms for big data privacy preservation, [22] uses support vector machine technique and the particle swarm optimization (PSO) algorithm in design of web crawler, etc.

We studied many research papers and found that though there is ample work existing for tweet sentiment analysis but there is scope of improvement. Some existing models are applied on structured datasets [15, 23, 24], some are applied on unstructured [16, 17] but size is not too big [18] or algorithm is non-scalable [2]. Hence, motivation is to optimize sentiment analysis process through an algorithm capable of handling big data.

3 Methodology

This section introduces a novel framework for query optimization during customer churn sentiment analysis. The proposed system includes five main steps as shown in Fig. 1. Data ingestion is the first and major step while working with big data. To carry out research over big pool of telecom tweets, we selected Apache Flume [25] for tweets extraction and stored them in Hadoop [26]. Data preprocessing consists of series of steps for preparing the raw data for further processing [27]. Twitter data are unstructured, consist of lots of noise and characters or symbols that are irrelevant. Once data are stored, preprocessing starts by tokenization, which is followed by the removal of stop words [28]. Lemmatization and eliminating retweets clean up data and then the output is forwarded to next phase for feature extraction. The number and type of features determine the complexity of training model [29]. We have used SentiWordNet [30] process for selecting features during feature extraction phase. SentiWordNet is a lexical resource used by many researchers in feature extraction phase to label text with numerical scores for positivity, negativity and objectivity. Furthermore, feature vector also comprises of hashtag, number of elongated words and number of punctuations. The final feature vector will be passed to sentiment classification phase. Now, training data are ready which we used to predict the query results. In matching phase, a test query is presented, preprocessed, features are extracted and finally matching is done with a classified database.

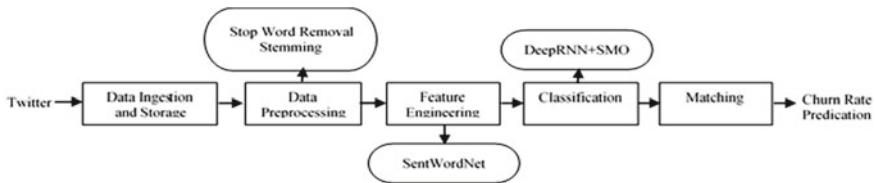


Fig. 1 Steps in the proposed model

4 Sentiment Classification

The proposed model does sentiment classification using 70% training and 30% test datasets. With DeepRNN, one value of input feature vector is picked up for processing. While moving forward, the network also maintains hidden units, which store history of all calculations. It does decision-making using current input and previously stored values [9, 15].

Although DeepRNN is good enough to handle big data but there training is a difficult task. Hence, we started training deep recurrent neural network using spider monkey optimization technique. Bansal et al. [10] designed a spider monkey optimization (SMO) algorithm based on the food searching habits of spider monkeys. The technique was inspired by two main characteristics of spider monkeys, namely, self-organization and division of labor. There are six phases in spider monkey. After initialization of control parameters, the possible solutions were characterized by the spider monkeys' position in their groups. Every iteration generates new position value, which was compared with the old position value to select the optimal value among both. Furthermore, their techniques proved successful in solving many global optimization problems. Literature shows that SMO helps in dimensionality reduction also [12, 20]. Hence, we used spider monkey for training weights at each stage to achieve optimization by reducing training time and finally improving performance metrics values of sentiment analysis models.

Let DeepRNN starts from b th layer at t th time. The input feature vector is represented as follows:

$$S^{(b,t)} = \{S_1^{(b,t)}, S_2^{(b,t)}, \dots, S_i^{(b,t)}, \dots, S_y^{(b,t)}\} \quad (1)$$

Output vector is expressed as

$$J^{(b,t)} = \{J_1^{(b,t)}, J_2^{(b,t)}, \dots, J_i^{(b,t)}, \dots, J_y^{(b,t)}\} \quad (2)$$

where, each element of input and output vectors is called as unit. In this classifier, i represents random unit of b th layer, and y signifies total units of b th layer. Moreover, random unit number and total units of $(b-1)$ th layer are expressed as j and E . Thus, weight of input transmission varies from $(b-1)$ th layer to b th layer is expressed as $W^{(b)} \in A^{y \times E}$, and recurrent weight of b th layer is expressed as $w^{(b)} \in A^{y \times y}$. Here, A symbolize weight set. Thus, the units of input vector are formulated in Eq. (3):

$$S_i^{(b,t)} = \sum_{z=1}^E p_{iz}^{(b)} J_z^{(b-1,t)} + \sum_{i'}^y x_{ii'}^{(b)} J_{i'}^{(b,t-1)} \quad (3)$$

To use spider monkey for training weights of DeepRNN, we initialized population B with total l spider monkeys, such that $1 \leq k \leq l$.

$$B = \{B_1, B_2, \dots, B_k, \dots, B_1\} \quad (4)$$

where, l is total solution, and B_k indicates the k th solution.

The mean square error (MSE) is computed by subtracting value of predicted output from actual output. The solution with less error is selected as best solution and is evaluated as,

$$E_c = \frac{1}{K} \sum_{g=1}^K J_v^{(b,t)} - \kappa_v \quad (5)$$

L_n signifies fitness of n th spider monkey, $J_v^{(b,t)}$ indicates output from DeepRNN classifier, and κ_v symbolizes estimated output. Finally, spider monkeys can update their positions [10] and then error corresponding to updated weights is computed using Eq. 5. Weights linked to the minimum error are employed for training DeepRNN. The optimum weights are attained in iterative manner until maximal iteration is accomplished.

5 Results and Analyses

We experimented with different values of training datasets—50, 60, 70, 80 and 90% respectively and compared results with GloVe-Deep Convolution Neural Network (DCNN) [14] in terms of four famous parameters, namely, accuracy, precision, recall, F-measure. Experimental results are represented in graphs in Fig. 2 and in Table 1.

The accuracy parameter of our hybrid model is more than DeepCNN for training set 50, 60, 70 and 90%; however, it is 0.88 for 80% training data, and DeepCNN value is 0.89 for 80% training data. F1-score and recall values are increasing with an increase in training data. Value of precision is less as compared with DeepCNN. The model has optimized the training time through self-organization and division of labor characteristics of SMO. Our model training time is 0.3 s, which is very less as compared with DeepCNN. Also, the test queries run faster and give accurate information during matching phase.

6 Conclusion

Our research aim is to work on query optimization techniques for unstructured big data. So, tweets are selected as they are heterogeneous in nature. This model is flexible to work with any size of data. The system empowers prediction process by performing query optimization during training and matching phase. Metaheuristics algorithms are getting famous these days to solve complex engineering problems. That is why we applied SMO as it is flexible and converge faster in search space. We evaluated the performance of our system by comparing results against existing deep convolutional neural networks (DeepCNN). We are working further to enhance the

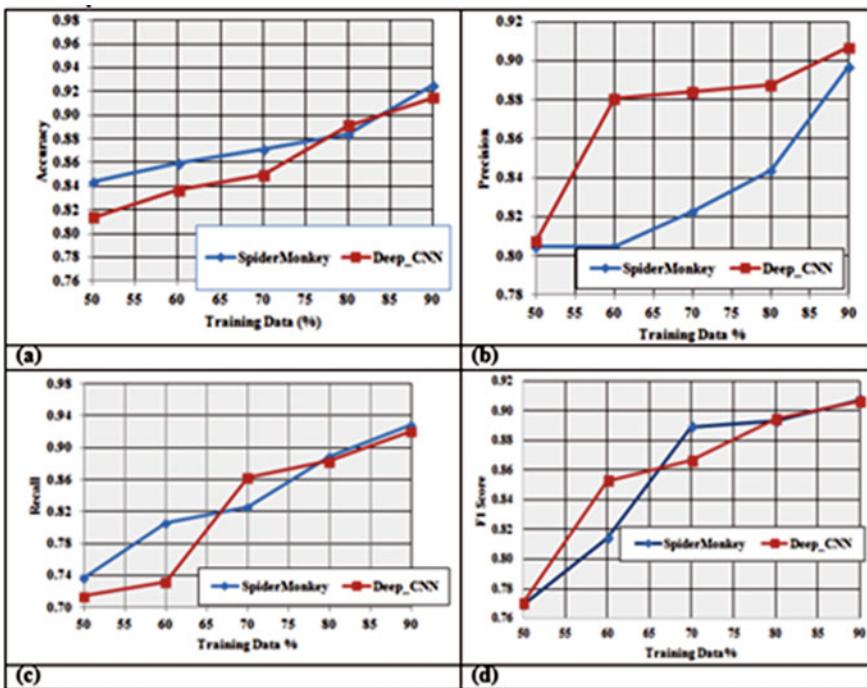


Fig. 2 Analysis based on sentiment classification using Twitter dataset in terms of **a** accuracy, **b** precision, **c** recall, **d** F1-measure

performance of our framework by using a hybrid of metaheuristic techniques. Once the complete framework for sentiment analysis is ready, it can be used in telecom sector to take opportune measures for improving the future client experience and keep away from customer churn.

Table 1 Comparative analysis

Training data	Accuracy			F1 score			Recall			Precision		
	Spider Monkey	Deep_CNN	Spider Monkey	Deep_CNN	Spider Monkey	Deep_CNN	Spider Monkey	Deep_CNN	Spider Monkey	Deep_CNN	Spider Monkey	Deep_CNN
50	0.84	0.81	0.77	0.77	0.74	0.71	0.80	0.71	0.80	0.81	0.80	0.81
60	0.86	0.84	0.81	0.85	0.81	0.73	0.80	0.73	0.80	0.88	0.80	0.88
70	0.87	0.85	0.89	0.87	0.83	0.86	0.82	0.86	0.82	0.88	0.82	0.88
80	0.88	0.89	0.89	0.89	0.89	0.88	0.88	0.88	0.84	0.89	0.84	0.89
90	0.92	0.91	0.93	0.93	0.93	0.92	0.90	0.92	0.91	0.91	0.90	0.91

References

1. N. Naga, P. Prithvi, Customer churn prediction using big data analytics (2016)
2. A. Kanavos, N. Nodarakis, S. Sioutas, A. Tsakalidis, D. Tsolis, G. Tzimas, Large scale implementations for twitter sentiment classification. *Algorithms* **10**, 1–21 (2017). <https://doi.org/10.3390/a10010033>
3. A. Alsaedi, M.Z. Khan, A study on sentiment analysis techniques of twitter data. *Int. J. Adv. Comput. Sci. Appl.* **10**, 361–374 (2019). <https://doi.org/10.14569/ijacsa.2019.0100248>
4. S. Shayaa, N.I. Jaafar, S. Bahri, A. Sulaiman, P. Seuk Wai, Y. Wai Chung, A.Z. Piprani, M.A. Al-Garadi, Sentiment analysis of big data: methods, applications, and open challenges. *IEEE Access* **6**, 37807–37827 (2018). <https://doi.org/10.1109/ACCESS.2018.2851311>
5. N.C. Dang, M.N. Moreno-García, F. de la Prieta, Sentiment analysis based on deep learning: a comparative study. *Electronics* **9**, 1–29 (2020)
6. A. Chugh, V. Sharma, C. Jain, Big data and query optimization techniques, in *Advances in Computing and Intelligent Systems. Algorithms for Intelligent Systems*, ed. by H. Sharma, K. Govindan, R. Poonia, S. Kumar, W. El-Medany (Springer, Singapore, 2020), pp. 337–345
7. K. Karanasos, A. Balmin, M. Kutsch, F. Özcan, V. Ercegovac, C. Xia, J. Jackson, Dynamically optimizing queries over large scale data platforms, in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data* (2014), pp. 943–954. <https://doi.org/10.1145/2588555.2610531>
8. 9.3. Deep recurrent neural networks—dive into deep learning 0.15.1 documentation, https://d2l.ai/chapter_recurrent-modern深深-rnn.html. Accessed 08 April 2018
9. M. Hermans, B. Schrauwen, Training and analyzing deep recurrent neural networks, pp. 1–9
10. J.C. Bansal, H. Sharma, S.S. Jadon, M. Clerc, Spider monkey optimization algorithm for numerical optimization. *Memetic Comput.* **6**, 31–47 (2014). <https://doi.org/10.1007/s12293-013-0128-0>
11. What is a recurrent neural network (RNN)—Arm, <https://www.arm.com/glossary/recurrent-neural-network>. Accessed 10 April 2018
12. N. Khare, P. Devan, C.L. Chowdhary, S. Bhattacharya, G. Singh, S. Singh, B. Yoon, SMO-DNN: spider monkey optimization and deep neural network hybrid classifier model for intrusion detection. *Electronics* **9** (2020). <https://doi.org/10.3390/electronics9040692>
13. S. Fong, S. Deb, X.S. Yang, How meta-heuristic algorithms contribute to deep learning in the hype of big data analytics. *Adv. Intell. Syst. Comput.* **518**, 3–25 (2018). https://doi.org/10.1007/978-981-10-3373-5_1
14. Z. Jianqiang, G. Xiaolin, Z. Xuejun, Deep convolution neural networks for twitter sentiment analysis. *IEEE Access* **6**, 23253–23260 (2018). <https://doi.org/10.1109/ACCESS.2017.2776930>
15. A.K. Ahmad, A. Jafar, K. Aljoumaa, Customer churn prediction in telecom using machine learning in big data platform. *J. Big Data* **6**, 28 (2019). <https://doi.org/10.1186/s40537-019-0191-6>
16. N.A. Vidya, M.I. Fanany, I. Budi, Twitter sentiment to analyze net brand reputation of mobile phone providers. *Procedia Comput. Sci.* **72**, 519–526 (2015). <https://doi.org/10.1016/j.procs.2015.12.159>
17. C. Virmani, D. Juneja, A. Pillai, Design of query processing system to retrieve information from social network using NLP. *KSII Trans. Internet Inf. Syst.* **12**, 1168–1188 (2018). <https://doi.org/10.3837/tiis.2018.03.011>
18. P. Sasikala, L. Mary Immaculate Sheela, Sentiment analysis of online product reviews using DLMNN and future prediction of online product using IANFIS. *J. Big Data* **7** (2020). <https://doi.org/10.1186/s40537-020-00308-7>
19. J. Mishra, B.K. Verma, Sentiment analysis with vector feature extraction and classification of social media dataset. *Int. J. Eng. Res. Comput. Sci. Eng.* **4**, 89–95 (2017)
20. K.K. Rao, G.L. Kumari, Y. Surekha, Heartbeat classification using the recurrent neural network based on the developed spider monkey-bird swarm optimization algorithm-proposed method. *Int. J. Adv. Sci. Technol.* **134**, 1–8 (2020). <https://doi.org/10.33832/ijast.2020.134.01>

21. S. Madan, P. Goswami, Nature inspired computational intelligence implementation for privacy preservation in MapReduce framework. *Int. J. Intell. Inf. Database Syst.* **13**, 191–207 (2020). <https://doi.org/10.1504/IJIIDS.2020.109455>
22. N. Kaushik, M.K. Bhatia, Information retrieval from search engine using particle swarm optimization (2020), pp. 127–140. https://doi.org/10.1007/978-981-15-0222-4_11
23. N. Kaushik, M.K. Bhatia, Various data classification technique on crime dataset. **X**, 84–92 (2020)
24. N. Kaushik, M.K. Bhatia, S. Rastogi, SVM and cross-validation using R studio. *Int. J. Eng. Adv. Technol.* **10**, 46–54 (2020). <https://doi.org/10.35940/ijeat.a1673.1010120>
25. A. Flume, Welcome to apache flume—apache flume, <https://flume.apache.org/>. Accessed 18 December 2020
26. Apache Hadoop, <https://hadoop.apache.org/>. Accessed 18 December 2020
27. R. Jony, Preprocessing solutions for telecommunication specific big data use cases (2014)
28. K.R. Jaideepsinh, R.S. Jatinderkumar, Stop-word removal algorithm and its implementation for Sanskrit language. *Int. J. Comput. Appl.* **150**, 975–8887 (2016)
29. R. Mansour, M.F.A. Hady, E. Hosam, H. Amr, A. Ashour, Feature selection for twitter sentiment analysis: an experimental study, in Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Springer, 2015), pp. 92–103. https://doi.org/10.1007/978-3-319-18117-2_7
30. S. Baccianella, A. Esuli, F. Sebastiani, SENTIWORDNET 3.0: an enhanced lexical resource for sentiment analysis and opinion mining, in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)* (2010), pp. 2200–2204

Lung Cancer Detection in Radiographs Using Image Processing Techniques



Bhawan Deep Singh, Chakshu Sharma, and Ashish Khanna

Abstract Cellular breakdown in the lungs tends to be a frequent cause of death in people all over the world. Individuals who are diagnosed with lung disease early on have a greater chance of survival. If the condition is diagnosed as predicted, the average 5-year survival rate for lung cancer patients rises from 14 to 49%. Despite the fact that computed Tomography (CT) is often more effective than X-ray. However, the problem tended to merge due to the time constraints in identifying the existence of malignant growth in the lungs, as well as the limited diagnostic techniques available. As a result, in CT pictures, a lung cancer identification system based on picture preparation is used to group the presence of cellular breakdown in the lungs. Using various update and division methods, the aim is to obtain more reliable results.

Keywords Image processing · Image enhancement · Feature extraction · Gabor filter enhancement

1 Introduction

The rising rate of mortality due to lung cancer is about to break the threshold of 170 lakh cases in whole world till 2030. The chances of survival in such type of cancer depends largely on the stage at which it is detected. Various diagnosing techniques seems to be exorbitantly priced and prolonged in duration such as Sputum Cytology, Scanning using Magnetic Resonance Imaging, Computerized Tomography, and Radiography of Chest. Including all the stages of cancer that patient goes through, early detection is found to be instrumental in recovery of a patient. There is utmost need to find the most effective and appropriate technology, which can highly impact the decision of life and death. Processing image digitally using CT scan can highly improve the chances of successful treatment. Lung Carcinoid

B. D. Singh (✉) · C. Sharma · A. Khanna
Maharaja Agrasen Institute of Technology, Delhi, India

A. Khanna
e-mail: ashishkhanna@mait.ac.in

Tumor exhibits different characteristics [1]. What actually results in lung cancer is nonuniform, uneven augmentation of cells present in tissues of lungs forming abnormalities in lung tissues are referred to as lung nodules varying in size of 1–30 mm [2]. This paper brings focus onto continuously emerging field of detection of lung nodules using CT scans by disintegrating the whole complex process by following three quintessential stages like Feature extraction, Segmentation of image, and Image enhancement stage. Using Artificial Neural Networks for classification of images of lung nodules and a clustering algorithm for performing segmentation will open the doors for most efficient and fast detection of lung cancer [3].

2 Methodology

The proposed work has been done in three stages, first data pre-processing then extraction of features, and finally classification of CT scans. The main software used for this study is MATLAB [4]. The flowchart of methodology is given in Fig. 1.

2.1 Preprocessing

The first step of image preprocessing was image enhancement, i.e., to polish the decipherability of details in the dataset for humans or to improve our results for image processing [5, 6]. Two techniques are used for image processing in this paper.

2.1.1 Enhancement

The two key types of image enhancement methods are the “spatial domain method” and the “frequency domain method” Regrettably, there is no broad hypothesis for determining what constitutes “great” image enhancement in terms of human perception. It is appropriate if it appears to be amazing! However, quantitative measures can determine appropriate methods for image enhancement when they are utilized for preprocessing. For our paper, Gabor filter, auto enhancement, and fast Fourier transform methods were used for image enhancement.

2.1.2 Segmentation

This process divides the image into sets of pixels. It is an essential process for all the tasks related to the analysis of images [7]. We used two image segmentation methods for our study; thresholding segmentation and watershed segmentation.

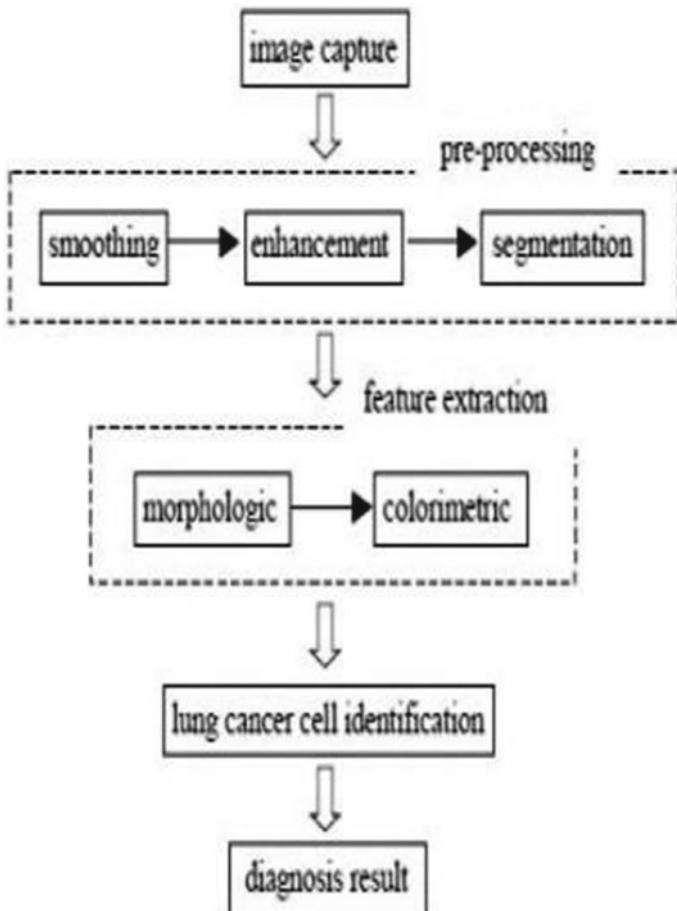


Fig. 1 Methodology

Thresholding Segmentation

There are three advantages of using thresholding segmentation: segmented images take up less space and are easy to manipulate.

Watershed Segmentation

It separates seeds demonstrating the presence of articles at explicit areas in the image. Those areas are activated as local minima and then the watershed algorithm is implemented.

2.2 Feature Extraction

Feature extraction is also referred to as dimensionality reduction. It increases accuracy and reduces the computational cost, therefore it is a very important step of data analysis. The obtained features are then used for classification. In our dataset, only three features were extracted: area, parameter, and eccentricity [8]. These features are thus defined below.

- (1) “Area: it is a scalar number that gives the genuine number of knob pixels. It is obtained by the summation of districts of a pixel in the image that is enrolled as 1 in the binary picture obtained.
- (2) Perimeter: It is a scalar value that gives the real number of the outline of the knob pixel.
- (3) Eccentricity: It is also referred to as roundness. The value of eccentricity is equal to 1 for circular shape and is short of what one for some other shapes.”

3 Results

The CT scans of patients with malignancy have productively completed the data and image preprocessing part. Three features are obtained from the data extraction process.

3.1 Image Enhancement Results

“Image enhancement” can also be defined as the process to improve the image quality for better perception by the human eye. MATLAB software was used for image enhancement. Three methods used for image enhancement are explained below.

3.1.1 Gabor Filter Enhancement Method

“It was proposed by Dennis Gabor [9]. It is a very useful tool for texture analysis because of its localization qualities in spatial domain and frequency domain”. The results obtained from the above-mentioned enhancement method are given in Fig. 2.

3.1.2 Auto Enhancement Method

It impulsively enhances and adjusts the brightness, color, and contrasts of the image to ideal magnitude. Statistical operations like mean and variance calculation play an important role in this method. Results of the auto enhancement method are shown in Fig. 3, where (a) is the original image and (b) is the enhanced image.

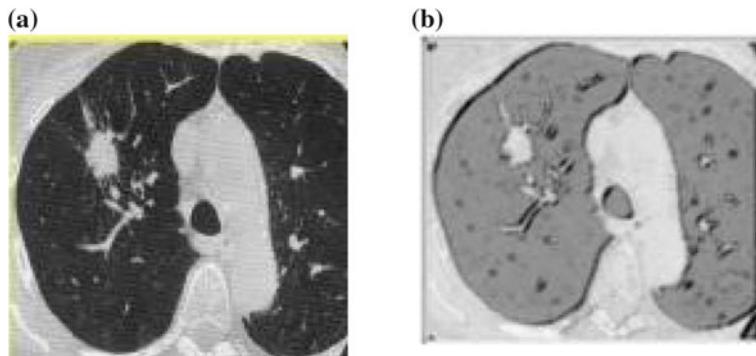


Fig. 2 Gabor filter enhancement method **a** initial image **b** optimized image

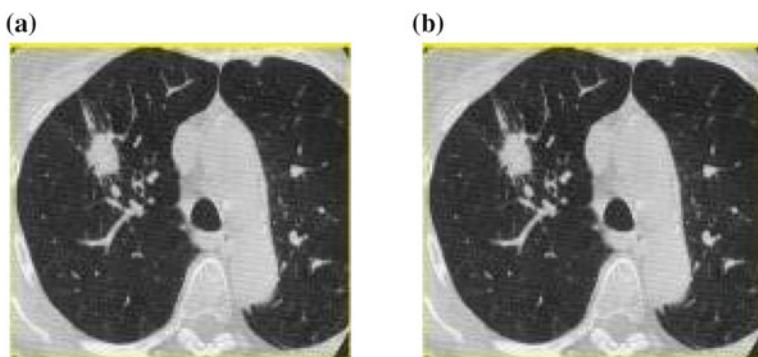


Fig. 3 Auto enhancement method **a** initial image **b** optimized image

3.1.3 Fast Fourier Transform Method

This method uses Fast Fourier transform (FFT) of the image to function [10]. Fast Fourier transform is the faster version of Discrete Fourier transform (DFT). The results obtained from FFT are presented in Fig. 4.

As obvious from the final enhanced images of CT scans obtained, the Gabor filter enhancement method is the best and most appropriate method for our study.

3.2 Segmentation Results

Segmentation divides images into different regions and it has many applications in health care or medical professions like detection of tumors or any abnormalities, tissue quantification and classification, etc.

In this paper, two segmentation techniques were used.

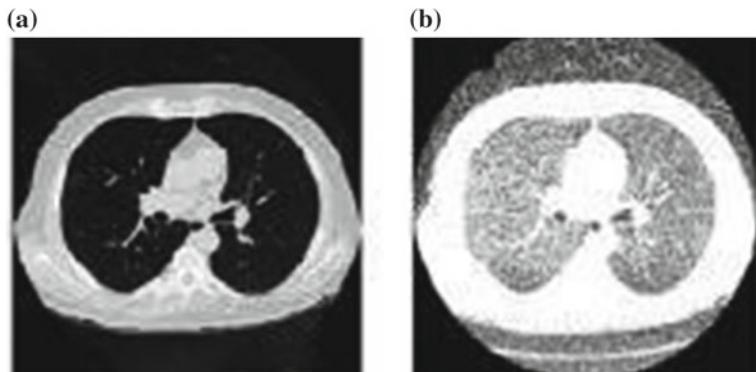


Fig. 4 FFT enhancement method **a** initial image **b** optimized image

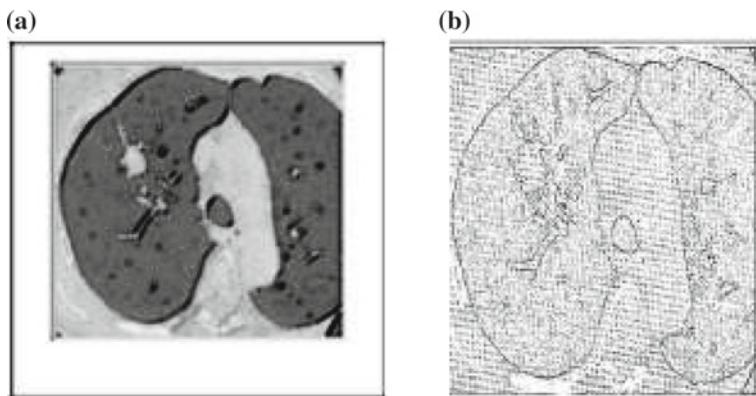


Fig. 5 By thresholding code in Gabor filter and segmentation **a** optimized image **b** segmented Image

3.2.1 Threshold Technique

It converts a grayscale image into a binary image, where the pixels are given two values, above or below the given specified threshold value. The results obtained from threshold segmentation are shown in Fig. 5, where (a) is the enhanced image and (b) is the segmented image.

3.2.2 Watershed Technique

Watershed segmentation technique has no smoothing and generalization properties, but it is perhaps the most troublesome picture handling technique because it separates

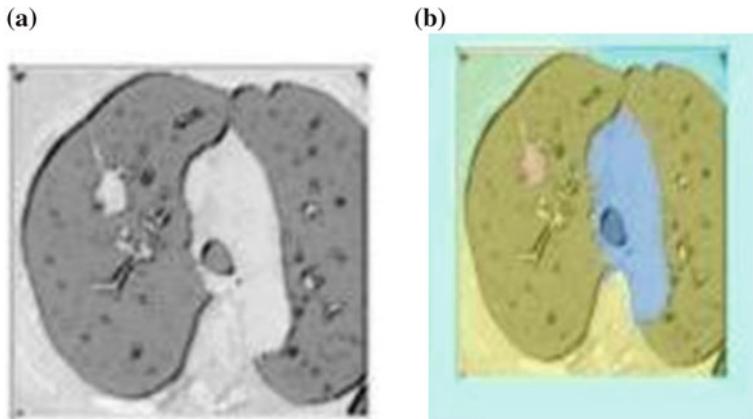


Fig. 6 By Marker-Watershed in Gabor filter and segmentation **a** optimized image **b** segmented image

touching objects in the image [11]. The results obtained from watershed segmentation are shown in Fig. 6, where (a) is the enhanced image and (b) is the segmented image.

The results obtained depict that watershed segmentation gives more accuracy than the threshold technique.

3.3 Feature Extraction Results

Feature extraction is also referred to as dimensionality reduction. It increases accuracy and reduces the computational cost, therefore it is a very important step of data analysis. The obtained features are then used for classification. In our dataset, only three features were extracted; area, parameter, and eccentricity. These features are thus defined below.

- (1) Area,
- (2) Perimeter, and
- (3) Eccentricity: It is also referred to as roundness. The value of eccentricity is equal to 1 for circular shapes and is short of what one for some other shapes.

4 Lung Cancer Staging

A disease's process refers to the degree to which a life-threatening condition has spread across the body. Organizing involves determining the tumor's size and penetration of adjacent tissue, as well as the existence or absence of metastases in lymph nodes or other organs. Since the cellular breakdown in the lungs therapies is designed

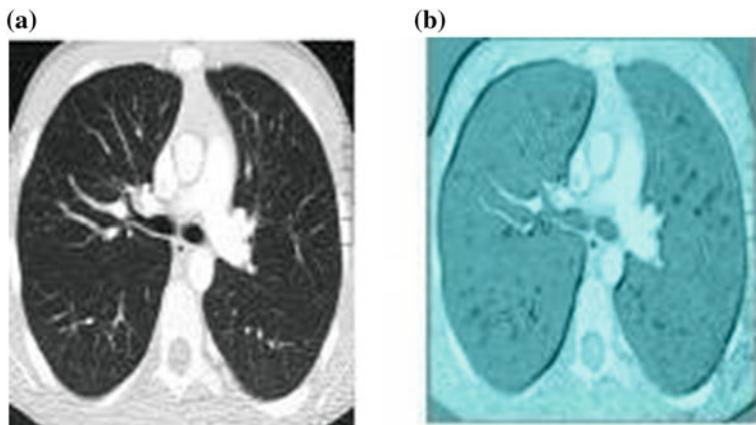


Fig. 7 Normal image: image segmentation **a** original image **b** segmented image

for particular stages, staging is important when determining how to treat a specific malignancy [12]. The staging of cancer is also critical in deciding a patient's prognosis, with higher stage tumors usually having a poorer prognosis than lower stage tumors.

4.1 Picture Segmentation Results for Normal Stages 1 and 2

See Fig. 7.

4.2 Results for Feature Extraction for Stages 1 and 2

See Figs. 8 and 9.

Stages I through IV, in order of severity:

- The malignancy is bound to the lung in stage I.
- The malignancy is bound to the chest in Stages II and III (with bigger and more obtrusive tumors delegated stage III).
- Stage IV malignant growth has spread from the chest to different pieces of the body.

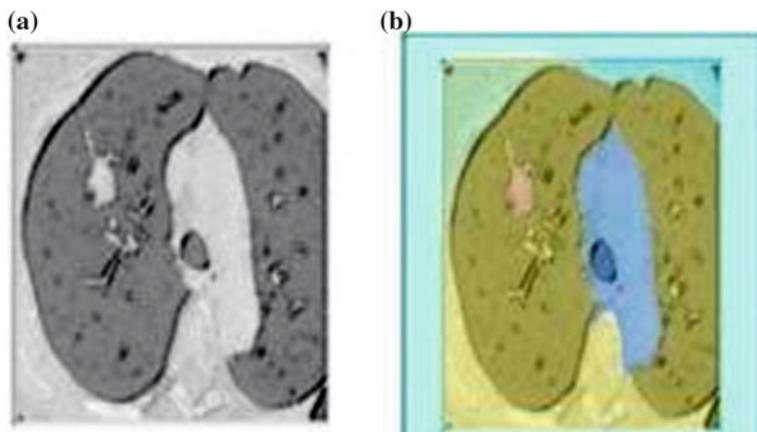


Fig. 8 Stage 1: image segmentation **a** original CT image **b** segmented image

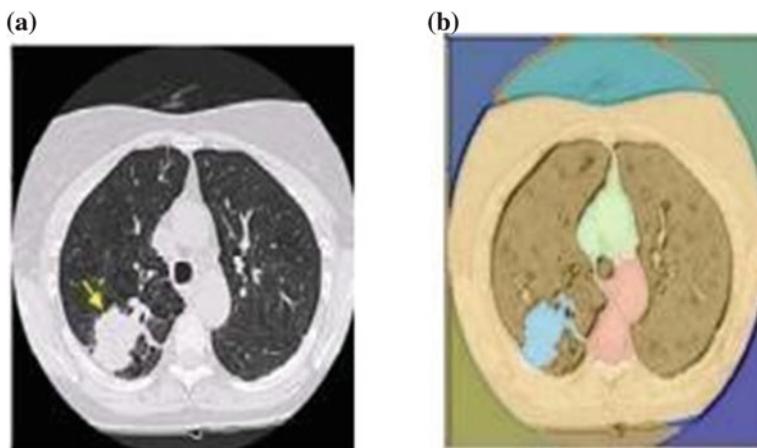


Fig. 9 Stage 2: image segmentation **a** original CT image **b** segmented image

5 Conclusion

According to arrangement of the disclosure of the harmful production cells in the lungs, cancer in the lungs is the most unsafe and widespread on the globe. This gives us the hint that in the pattern of acknowledgment, this illness plays a vital and central capacity to stay away from genuine stages and to minimize its rate of transport in the nation. The work was divided into three levels or categories to improve accuracy: Image Enhancement, Image Segmentation, and Features Extraction.

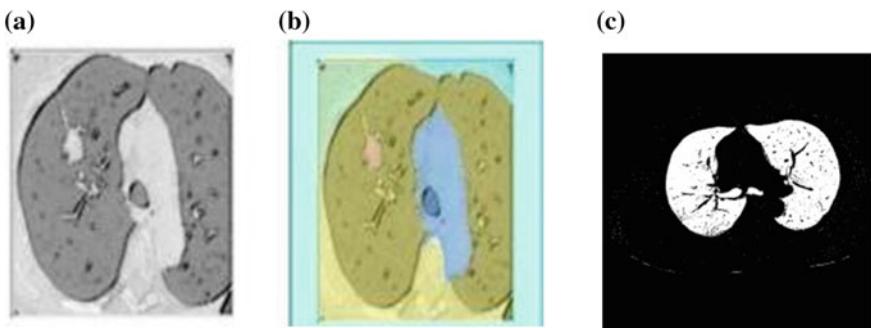


Fig. 10 Results of image segmentation

6 Comparison with Previous Work

In the existing computer-aided design-based nodule detectors, the lungs are physically set apart by the radiologist, which is a dull and tedious undertaking. In the proposed work, the lungs are automatically segmented from the CT scans with no client intercession. Nodules can have different regular and irregular shapes and sizes. Some current strategies use a couple of shape templates to detect the nodule, however, the proposed algorithm is independent of the node's shape and size as shown in Fig. 10. Left side are the results of segmentation using the watershed algorithm, whereas right side are the results of existing CAD systems. Therefore, clearly, the details are better visible in our proposed work, as perceptible by the human eye.

References

1. A. Jemal et al., Cancer Statistics, 2005. CA. Cancer J. Clin. (2005), canjclin.55.1.10
2. D.T. Lin, C.R. Yan, W.T. Chen, Autonomous detection of pulmonary nodules on CT images with a neural network-based fuzzy system. Comput. Med. Imaging Graph. (2005). <https://doi.org/10.1016/j.compmedimag.2005.04.001>
3. J.E. Bibault, P. Giraud, A. Burgun, Big Data and machine learning in radiation oncology: state of the art and future prospects. Cancer Lett. (2016). <https://doi.org/10.1016/j.canlet.2016.05.033>
4. L. Keviczky, R. Bars, J. Hetth  ssy, C. B  ny  sz, Introduction to MATLAB. Adv. Textb. Control Signal Process. (2019). https://doi.org/10.1007/978-981-108321-1_1
5. A. El-Bazl, A.A. Farag, R. Falk, R. La Rocca, Automatic identification of lung abnormalities in chest spiral CT scans (2003), <https://doi.org/10.1109/icassp.2003.1202344>
6. B. Van Ginneken, B.M. Ter Haar Romeny, M.A. Viergever, Computer-aided diagnosis in chest radiography: a survey, IEEE Trans. Med. Imaging (2001), <https://doi.org/10.1109/42.974918>
7. S. Wang, R.M. Summers, Machine learning and radiology. Med. Image Anal. (2012). <https://doi.org/10.1016/j.media.2012.02.005>
8. E. Dougherty, S. Beucher, and F. Meyer, The morphological approach to segmentation: the watershed transformation, in *Mathematical Morphology in Image Processing* (2019)
9. V.S.N. Prasad, J. Domke, *Gabor Filter Visualization* (University of Maryland, 2005)

10. S. Lin et al., FFT-based deep learning deployment in embedded systems (2018), <https://doi.org/10.23919/date.2018.8342166>
11. H.T. Nguyen, M. Worring, R. Van den Boomgaard, Watersnakes: energy-driven watershed segmentation. IEEE Trans. Pattern Anal. Mach. Intell. (2003). <https://doi.org/10.1109/TPAMI.2003.1182096>
12. K. Suzuki, J. Shiraishi, H. Abe, H. MacMahon, K. Doi, False-positive reduction in computer-aided diagnostic scheme for detecting nodules in chest radiographs by means of massive training artificial neural network. Acad. Radiol. (2005). <https://doi.org/10.1016/j.acra.2004.11.017>

COVID-19 Spread: A Demographic Analysis



Yashi Srivastava, Pooja Khanna, Sachin Kumar, and Pragya

Abstract The relationship of COVID-19 cases with growth rate, literacy, and other data points that describe people of a country or their living standards, might be insightful in making predictions and decisions in the long run. Leading Health Boards across the world have analyzed various related statistics with a conclusion that COVID-19 infection is going to stay here for some upcoming time. Making a thoughtful and informed decision after analyzing the situation is much better than doing trial and errors and playing with the health of people, thus, to make a relationship analysis between growth and development parameters of the country to the cases and deaths due to virus reported by that country, following main data points were collected for different countries across the globe: Literacy Rate, GDP, Percentage of GDP spent on Health, Total number of Corona Cases reported, Total Cases per million of population, Population density, Gross Income per capita, Number of internet users, Total Deaths due to COVID-19 Virus, Percentage of population below poverty line (BPL), and Health workers density. Work presents a relational demographic analysis with K-means clustering on parameters mentioned to establish a correlation between infection spread and associated parameters, with certain exceptions and reasons for it. The work proposed clearly outlines under-rated parameters that potentially impact the spread of COVID-19 spread, majorly low literacy, machine-dependent lifestyle, low economic stability, high population density, large migrants, limited healthcare infrastructure, and less gross national income per capita raised insecurities and contributed to infection spread. However, exceptions to the above exist, citing reasons such as

Y. Srivastava · P. Khanna

Department of Computer Science, Amity University Lucknow Campus, Lucknow, India
e-mail: pkhanna@lko.amity.edu

S. Kumar (✉)

Department of Electronics & Communication, Amity University Lucknow Campus, Lucknow, India
e-mail: skumar3@lko.amity.edu

Pragya

Department of Chemistry, MVD College, Lucknow University, Lucknow, India

stringent measures such as complete lockdown, environmental conditions, and effectiveness of diversified vaccinations already existing on COVID-19 such as BCG, all under research.

Keywords COVID-19 · Population density · BPL · Literacy · K-means

1 Introduction

Novel COVID-19 infection is a disease caused by Coronavirus, which is defined as Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2). The first case of Coronavirus disease was identified in December'19 in the city of Wuhan in China. The horrifying fact of this virus infection is that it spreads too fast. Since the first inception of cases in December 2019, till May 21, 2020, there are 5.01 million reported cases globally. With flu-like symptoms, the disease was identified as pneumonia initially.

When the cluster occurred in Wuhan, research was carried out, and Novel COVID-19 Virus was identified. Within one month of the first case, the virus became a global threat, as on January 13, 2020, the first case was identified out of China, in the country of Thailand. It was late January 2020 when World Health Organization (WHO) confirmed that COVID-19 transmits from human-to-human transmission. By that time, there were already more than 7,000 cases identified globally; the virus had spread across 19 nations of the world. Then WHO declared the COVID-19 virus as PHEIC (Public Health Emergency of International Concern).

Common Symptoms of the COVID-19 virus are fever, cough, and breathlessness. The disease caused by COVID-19 was not deadly in most of the cases diagnosed, the comorbidity was a worrisome sign as the immune system weakens with the presence of multiple medical conditions.

Commonly the virus is spread by close contact with an infected person, where while talking, sneezing the water droplets may fall on other people. The less spread of virus is touching surfaces touched by infected beings.

COVID-19 virus was declared a pandemic on March 11, 2020, by the World Health Organization. Cambridge Dictionary defines pandemic as a disease which affects people around the world at the same time. COVID-19 cases are seeing an exponential increase ever since they evolved. The scare of mass spread can be justified, by the fact India took 97 days to reach the first 50,000 cases, and just 14 days to reach 1,00,000 cases from 50,000 cases, even after one of the biggest lockdowns being implemented across the globe. Recently, WHO warned that people should learn to live with COVID-19 virus as it may never go away [1–3].

The population of developed countries has a good lifestyle and diet due to higher per capita income. Thus, due to a good diet, they tend to develop a good immune system. Developed nations have better healthcare facilities, and medical personnel per thousand population are higher due to the fact that a good amount of GDP is spent on the health sector. If the case of developing nation is considered, the population over

there is found to be under more stress, however, the living standard is nonuniform, i.e., a proportion of population might have as good living standards and eating habits as a developed nation, and another proportion might not have great living standards. In developing nations, the health setup, if not great, is good enough to deal with normal patients daily, however, it may flunk in case of a pandemic. It may also be noted that people in developed countries live in a sparser setup than their counterparts. Discussing under-developed nations, the health facilities, eating habits, nutrition in diet, population density, all such factors, and availability of resources, go for a toss at once. The situation in the day-to-day life of under-developed nations in normal situations is bad, and in case of a pandemic it tends to be worse.

The chapter is organized as follows; Sect. 2 presents the motivation for taking up the work, Sect. 3 provides the methodology proposed for the work, Sect. 4 presents the clustering implementation of suspected COVID-19 spread parameters, Sect. 5 explores the results concluded with a discussion and finally paper concludes with major potential reasons for COVID-19 spread.

2 Motivation

With growing cases day by day, the world's best health infrastructure has failed to handle COVID-19 virus spread, resulting in the Health Minister of a few European countries, even resigning, and accepting their failure. It is extremely important to analyze the spread of virus based on socioeconomic strata and behavioral characteristics of people around the globe, to make an informed and well-understood decision, instead of making blind shots.

It was even observed that the lockdown implemented in a few countries gave good results. However, the situation worsened as soon as the lockdown was uplifted. Even, it was observed that lockdowns led to the economic crisis, making people starve for 2 square meals. Lockdowns served as a buffer period to let the governments prepare for the upcoming crisis. In these times, almost all countries around the globe made significant progress and prepared themselves.

It was observed that a few countries were following the footsteps of other countries, who have been able to curb COVID-19 virus in their nation. But this approach cannot be followed blindly as the situation varies from nation to nation. A methodology good for one nation might prove deteriorating for another nation. So, it is well required for the government, analyze well the demographics of their nation, and take decisions accordingly [4–7].

Across the globe, diversified models have been designed and are being worked upon to predict reasons for the steep rise in infection and finding potential parameters that might slow down the growth. Zhang [8] presented a data analytics model for estimation of novel coronavirus (COVID-19) reproduction and probable outbreak size on the diamond princess cruise ship. Yang et al. [9] trained an LSTM model initially designed for SARS (2003) data because the times series data of SARS is complete and could be employed for COVID-19 spread with similar features. Giuliani

[10] et al. have developed statistical models with machine learning classification for predicting epidemic COVID-19 infection spread and tried to identify potential reasons for the high steep spread. A number of researchers have put forth diversified prediction models based on a number of parameters for forecasting COVID-19 spread in different geographical locations, prediction models are generally based on mathematical models with applied learning algorithms and clustering techniques for epidemic prediction based on time series [11, 12]. Time series analysis logistic is often used in regression fitting for easy and accurate calculation. Infection is usually characterized by a slow rise at the start, steep rise in the middle of the incidence curve, and a slow rise at the end of the outbreak, Wu et al. [13] precisely modeled a logistic model, the generalized model so developed predicted the reported number of COVID-19 infections with accuracy, the logistic model could also give upper and lower bounds of our scenario predictions. Taylor S. J. et al. put forth that SARS and COVID-19 are very different in many ways, such as incubation rate, so the studies conducted with data later may not be reliable [14].

The purpose of this study is to derive logical conclusions, based on the demographics of the country, to provide an intuitive prediction about the fact, how worse can be this COVID-19 spread across different nations.

3 Methodology

Analyzing the relationship of COVID-19 cases with growth rate, literacy, and other data points that describe people of a country or their living standards, might be insightful in making predictions and decisions in the long run. It has been analyzed by the leading Health Boards that, across the world, the corona is going to stay for some upcoming time. So, in this situation, making a thoughtful and informed decision after analyzing the situation is much better than doing trial and error and playing with the health of people.

Thus, to make a relationship analysis between growth and development parameters of the country to the cases and deaths due to virus reported by that country, following main data points were collected for different countries around the globe:

1. Literacy rate,
2. GDP,
3. Percentage of GDP spent on Health,
4. Total number of corona cases reported,
5. Total cases per million of population,
6. Population density,
7. Gross income per capita,
8. Number of internet users,
9. Total deaths due to COVID-19 virus,
10. Population % below the poverty line
11. Health workers density.

The method adapted in analyzing relationships will be described. The data points collected were country to value figures, but it was scattered over multiple files. The first task was compiling them into a large single file to efficiently analyze relations between various points. This could have been a tedious task if done manually, since the data was spread across 5 files, and there were 180+ countries in every file. Therefore, with multiple csv files in hand, Python 3 was picked up as a tech stack to combine and analyze the data. Since the values were collected from various sources, data pre-processing and cleaning were also required.

The key tasks were as follows:

1. Importing Data and Merging Data in Workspace—In this Pandas library was imported. After that, all data csv files were read and viewed to verify that the read is successful. Then, multiple files were merged into one (Fig. 1).
2. Cleaning, pre-processing data—This included removing “,” from the number of cases and other numerical values to let the Python interpret the field as a number instead of a string, and thus analysis could be performed easily. Performing pre-processing for country names, since some data had abbreviated names, and others had expanded forms. Dropping and/or setting rules to ignore NaN fields was established in this part.

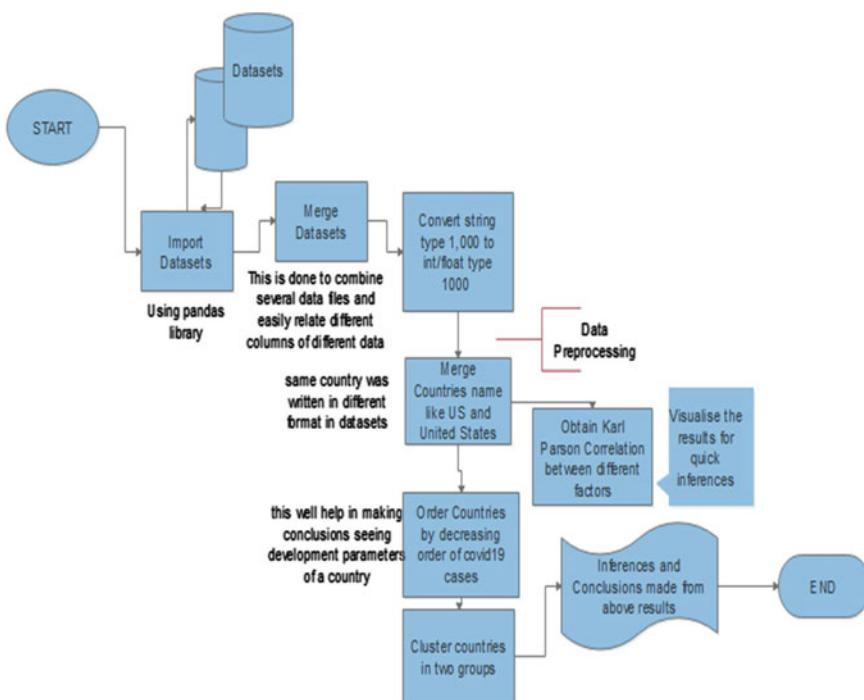


Fig. 1 Workflow

3. Correlation—Karl Pearson's Correlation was obtained between all the data points of the merged dataset. The correlation matrix was further visualized with key parameters as one of the axis, the correlation with that parameter on another axis. Also, data points were grouped into 2 and plotted, to see the dependency and relationship.
4. Making Inferences—Seeing the visualizations and the corresponding correlation matrix, conclusions were made and the probable reasons for such inferences were emphasized upon. This will be discussed in detail in the next section.

Analysis was performed taking into account potential factors, contributing to the spread of COVID-19. Factors included the following:

Population of any country plays a vital role in the creation of policy by the government of the nation. The budget of a nation and the proportionate division of the population in various sectors depends largely on the population of a nation. As, an intuitive fact, policies of nations to an extent describe the response of nations to situations of crisis.

Population Density plays a crucial role in the spread of communicative disease in society. Higher the population density, higher the chances of virus spread. Hence, for observing a pattern in covid19 cases, the population density was taken in account.

Income average of people of a country describes their living standard, which in turn is somewhat an evidence onto the quality of food intake and capability to access health facilities. Hence, the income was considered, as it may play a crucial role in relating mortality rate due to viruses.

Literacy Rates and more internet users can be an intuitive sign of a higher sense of awareness and rationality. Response to the crisis of an educated audience probably would be better than the rest. Higher literacy rates might have good practices like taking precautions and having healthy habits.

Healthcare Facilities play an important role to curb the spread and the ability to treat COVID-19 patients preventing casualties. Thus, healthcare facilities were taken into consideration to see how good recovery rates are, and what the extent of spread of virus is.

Corona Virus Statistics were taken to see the relations of the above-mentioned features with these statistics, like active cases, total cases, death, recoveries, etc.

4 Clustering: Nations and COVID-19 Effect

Similar countries were clustered based on demographics to analyze the COVID-19 effect on similar groups. However, the Poverty Data was not available for as large as 25% of nations, thus, the results obtained cannot be given as theoretical proof. And, moreover, some other data points also suffered missing values. In clustering including BPL Population could have given ambiguous results, thus, the feature BPL Population was dropped. BPL Population is a significant indicator of the progress of the Country, the model would have been wrongly fed, if BPL Population were

Group1
 ['Peru', 'Armenia', 'Belarus', 'Djibouti', 'Brazil', 'Maldives', 'Russia', 'Sao Tome and Principe', 'Moldova', 'Ecuador', 'Iran', 'Dominican Republic', 'Turkey', 'North Macedonia', 'Gabon', 'Bolivia', 'Serbia', 'Cabo Verde', 'South Africa', 'Mexico', 'Colombia', 'Azerbaijan', 'Equatorial Guinea', 'Honduras', 'Bosnia and Herzegovina', 'Kazakhstan', 'Guinea-Bissau', 'Ukraine', 'Pakistan', 'Afghanistan', 'El Salvador', 'Guatemala', 'Bangladesh', 'Albania', 'Tajikistan', 'Iraq', 'Montenegro', 'Bulgaria', 'Eswatini', 'Mauritania', 'Suriname', 'Haiti', 'Ghana', 'Cameroon', 'Guinea', 'Senegal', 'Malaysia', 'Algeria', 'Dominica', 'India', 'Morocco', 'Philippines', 'Nicaragua', 'Georgia', 'Lebanon', 'Nepal', 'Jamaica', 'Grenada', 'Comoros', 'Guyana', 'Cuba', 'Paraguay', 'Uzbekistan', 'Indonesia', 'Sierra Leone', 'Iberia', 'Jordan', 'Tunisia', 'Mali', 'Sri Lanka', 'Bhutan', 'Nigeria', 'Zambia', 'Kenya', 'Libya', 'Togo', 'Mongolia', 'China', 'Belize', 'Chad', 'Madagascar', 'Rwanda', 'Thailand', 'Benin', 'Burkina Faso', 'Niger', 'Ethiopia', 'Malawi', 'Botswana', 'Zimbabwe', 'Fiji', 'Mozambique', 'Timor-Leste', 'Uganda', 'Namibia', 'Tanzania', 'Burundi', 'Cambodia', 'Myanmar', 'Angola', 'Vietnam', 'Lesotho', 'Papua New Guinea']

 Group2
 ['Qatar', 'Singapore', 'US', 'Luxembourg', 'Iceland', 'Belgium', 'Sweden', 'Ireland', 'UK', 'UAE', 'Isle of Man', 'Switzerland', 'Netherlands', 'Cayman Islands', 'Canada', 'France', 'Germany', 'Denmark', 'Israel', 'Austria', 'Norway', 'Finland', 'New Zealand', 'Australia', 'Japan']

 Group3
 ['Bahrain', 'Andorra', 'Chile', 'Kuwait', 'Spain', 'Panama', 'Oman', 'Italy', 'Saudi Arabia', 'Portugal', 'Channel Islands', 'Bermuda', 'Estonia', 'Malta', 'Romania', 'Aruba', 'Cyprus', 'Poland', 'Argentina', 'Slovenia', 'Lithuania', 'Latvia', 'Croatia', 'Hungary', 'Costa Rica', 'Barbados', 'Greece', 'Antigua and Barbuda', 'Mauritius', 'Uruguay', 'Greenland', 'French Polynesia', 'Somalia', 'Seychelles', 'Trinidad and Tobago', 'New Caledonia']

Fig. 2 Clustering on selected parameters

included. For the features with lesser, like up to 5% missing values, missing values were filled with the mean of the feature. The three clusters of countries are present in Fig. 2.

The data were clustered in 3 clusters using K-Means Algorithm. The parameters that were taken into consideration while clustering was

["Population", "Surface area", "Population density", "Gross national income, Atlas method", "Gross national income per capita, Atlas method", "Tot Cases/\n1M pop", "GDPSpentonHealth", "Physician per 1k", and "Nurses per 1k"].

Specialist/1Lac was dropped due to unavailability for many countries.

From Fig. 3, the correlation between Cases Occurred and BPL population may be inferred as the disease occurs majorly in countries that are on the upper side of development statistics. But, it is also required to understand that these countries, with higher development numbers, have much higher migrant populations, making them more prone to virus communication from a foreign land. It may also be observed that the more tests per million population were conducted in these countries with good development numbers.

The second inference which could be made is that population density also played a major role in the increased number of COVID-19 cases. It can be observed from Fig. 4 that India, UK, Italy, and Germany are in the top 10 in the number of cases that have occurred. Other countries in the list of the highest number of cases also have significantly more population than the countries on the lower side of the list, barring some exceptions.

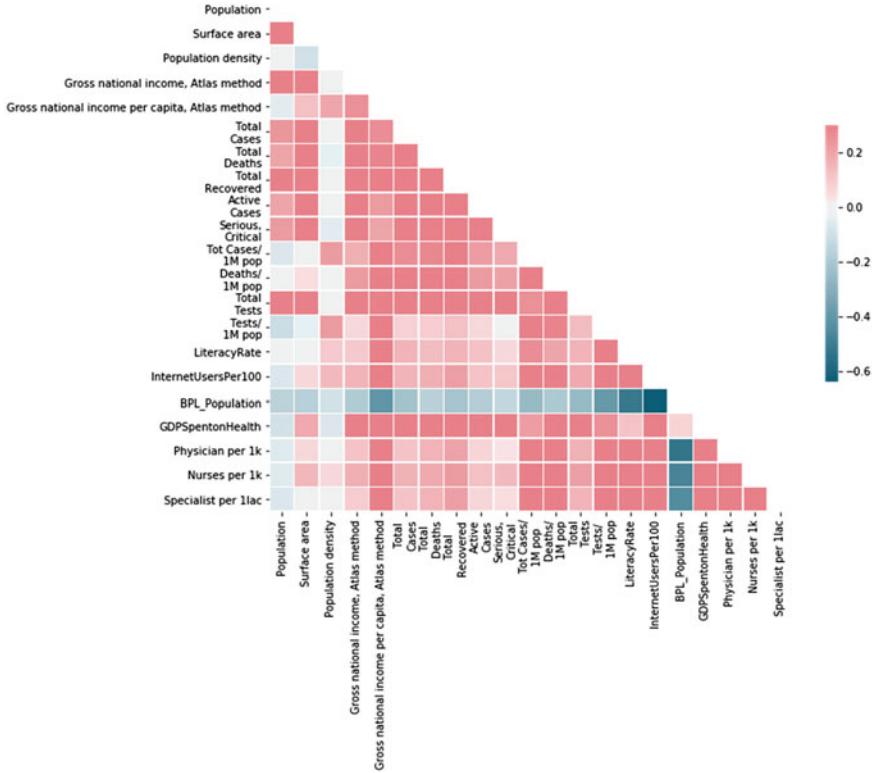


Fig. 3 Correlation between factors and case statistics

5 Results and Discussion

If we try to decipher the pattern in the cases observed across different countries in Fig. 4, it is observed that population density was one of the key factors in a high number of cases, provided the nation is either developing or developed. India, UK, Italy, Germany, and Pakistan are instances of such observation. The better economic condition of the country was responsible for the initial spread of the virus. It is so because such nations have higher numbers of people frequently flying in and out of such nations. Thus, in the initial days, when travel restrictions were not in place as such, the frequent travelers posed higher cases in such nations. Also, the scare of the virus was such that in the period of early cases, employees of various countries and international students started moving to their homeland.

While developed nations like the US, UK, Germany, and Russia were instances of former observation, developing Asian countries like India and Pakistan were instances of later observations.

The nations that were able to conduct higher numbers of tests per million of the population also faced higher numbers of cases. However, the increased testing rate

Country	Population	Surface area	Population density	Gross national income, Atlas method	Gross national income per capita, Atlas method	Total Cases	Total Deaths	Total Recovered	Active Cases	...	Total Tests	Tests/1M pop	Li
US	327.200012	9831.500000	36.0	20636.300781	63080.000000	2182951.0	118283.0	889866.0	1174802.0	...	25259076.0	76329.0	
Brazil	209.500000	8515.799805	25.0	1915.300049	9140.000000	891556.0	44118.0	464774.0	382664.0	...	1628482.0	7664.0	
Russia	144.500000	17098.300781	9.0	1501.699951	10230.000000	545458.0	7284.0	294306.0	243868.0	...	15300000.0	104843.0	
India	1352.599976	3287.300049	455.0	2727.899902	2020.000000	344407.0	9921.0	180460.0	154026.0	...	5921069.0	4292.0	
UK	66.500000	243.600006	275.0	2777.399902	41770.000000	296857.0	41736.0	148428.5	148428.5	...	6866481.0	101169.0	
Spain	46.700001	505.899994	94.0	1371.000000	29340.000000	291189.0	27136.0	145594.5	145594.5	...	4826516.0	103232.0	
Italy	60.400002	301.299988	205.0	2038.400024	33730.000000	237290.0	34371.0	177010.0	25909.0	...	4648825.0	76884.0	
Peru	32.000000	1285.199951	25.0	207.100006	6470.000000	232992.0	6860.0	119409.0	106723.0	...	1376478.0	41773.0	
Iran	81.800003	1745.199951	50.0	441.000000	5470.000000	189876.0	8950.0	150590.0	30336.0	...	1269194.0	15119.0	
Germany	82.900002	357.600006	237.0	3905.300049	47090.000000	188044.0	8885.0	173100.0	6059.0	...	4694147.0	56034.0	
Turkey	82.300003	785.400024	107.0	858.099976	10420.000000	179831.0	4825.0	152364.0	22842.0	...	2674203.0	31723.0	
Chile	18.700001	756.700012	25.0	274.799988	14670.000000	179436.0	3362.0	148792.0	27282.0	...	858958.0	44950.0	
France	67.000000	549.099976	122.0	2752.000000	41080.000000	157372.0	29436.0	73044.0	54892.0	...	1384633.0	21215.0	
Mexico	126.199997	1964.400024	65.0	1159.000000	9180.000000	150264.0	17580.0	113006.0	19678.0	...	415097.0	3221.0	
Pakistan	212.199997	796.099976	275.0	337.100006	1590.000000	148921.0	2839.0	56390.0	89692.0	...	922665.0	4181.0	
Saudi Arabia	33.700001	2149.699951	16.0	727.799988	21600.000000	132048.0	1011.0	87890.0	43147.0	...	1126653.0	32385.0	

Fig. 4 Data of countries with factors and with cases sorted in decreasing order

is not a cause of the widespread virus but a true reflection of the situation in the nation. US, Spain, Russia, Italy, Peru, Iran, Saudi Arabia, Chile, and Germany are instances of such observation. But this is the significance of the fact, the nations with commendable health care infrastructure, were able to almost diagnose every present case in the region.

There might be an argument, that the observations made here have exceptions, and a few nations lower in the tally of Fig. 4, exhibit the same demographics as discussed above. But it has to be a noted fact, the spread of the virus was also controlled due to immune, awareness, and eating habits of people, which cannot be quantified, so a few exceptions are well justified. The other reason might be the fact, the nations on the lower side of the tally might have to face the peak of cases yet. An example of such an instance is the situation of India and Pakistan, around late April. At that time while developed nations were experiencing high crises, India, Nepal, and Pakistan had a much smaller number of cases, but in around 2 months the table turned around. So, if the country is facing lower cases than the expected, such nations have to be highly careful in the times ahead and take full precautions, so they are able to flatten the curve and are safe from the second wave.

6 Conclusion

It can be concluded that a combination of high development numbers and high population are signs of a higher number of COVID-19 cases. But an inferring fact like COVID-19 is restricted to rich or more developed countries, maybe entirely wrong to make. To not make any such inference, the BPL Population of countries with a higher number of corona cases was observed. It was observed that, for countries where BPL population data is available, the cases are more where BPL Population, Population Density, and Development Number together are more. Work proposed clearly outlines under-rated parameters that potentially impact the spread of COVID-19 spread, majorly low literacy, machine-dependent lifestyle, low economic stability, high population density, large migrants, limited healthcare infrastructure, and less gross national income per capita raised insecurities and contributed to infection spread. However, exceptions to above exist, citing reasons such as stringent measures such as complete lockdown, environmental conditions, and effectiveness of diversified vaccinations already existing on COVID-19 such as BCG, all under research.

The exceptions in some parts of the world may also be described at case discovery early, where the research on COVID-19 lacked and there was a lack of medical data and preparedness. The countries performing better than similar counterparts need to be highly vigilant of the situation, as they might not have experienced the worst so far.

References

1. <https://data.worldbank.org/indicator/Population> and area numbers of countries, Literacy Rates, Health Facilities, Economic Numbers, Internet Users
2. <https://www.worldometers.info/coronavirus/> Covid19 Numbers reported till 16th June, 2020, 12:00 PM IST
3. World Health Organization, Coronavirus disease (COVID-2019) situation reports (2020), <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports/>. Accessed 04 May 2020
4. H. Cheng, S. Li, C. Yang, Initial rapid and proactive response for the COVID-19 outbreak—Taiwan’s experience. *J. Formos. Med. Assoc.* **119**(4), 771–773 (2020)
5. S. Park, G.J. Choi, H. Ko, Information technology-based tracing strategy in response to COVID-19 in South Korea—privacy controversies. *JAMA* (2020)
6. L. Gardner, Update January 31: modeling the spreading risk of 2019-nCoV. Johns Hopkins University Center for Systems Science and Engineering. Published 2020. <https://systems.jhu.edu/research/publichealth/ncov-model-2>. Accessed 20 Feb 2020
7. C.J. Wang, C.Y. Ng, R.H. Brook, Response to COVID-19 in Taiwan: Big Data analytics, new technology, and proactive testing. *JAMA* **323**(14), 1341 (2020)
8. S. Zhang, M. Diao, W. Yu, L. Pei, Z. Lin, D. Chen, Estimation of the reproductive number of novel coronavirus (COVID-19) and the probable outbreak size on the diamond princess cruise ship: a data-driven analysis. *Int. J. Infect. Dis.* **93**, 201–214 (2020). <https://doi.org/10.1016/j.ijid.2020.02.033>

9. Z. Yang, Z. Zeng, K. Wang, S. Wong, W. Liang, M. Zanin et al., Modified seir and ai prediction of the epidemics trend of COVID-19 in China under public health interventions. *J. Thorac. Dis.* **12**(2), 165–174 (2020). <https://doi.org/10.21037/jtd.2020.02.64>
10. D. Giuliani, M.M. Dickson, G. Espa, F. Santi, Modelling and predicting the spatio-temporal spread of coronavirus disease 2019 (COVID-19) in Italy (2020). <https://doi.org/10.2139/ssrn.3559569>
11. D.P.F. Fanelli, Analysis and forecast of COVID-19 spreading in China, Italy and France. *Chaos Solitons Fractals* **134**, (2020). <https://doi.org/10.1016/j.chaos.2020.109761>
12. M. Yousaf, S. Zahir, M. Riaz, Statistical analysis of forecasting COVID-19 for up-coming month in Pakistan. *Chaos Solitons Fractals* **109926** (2020). 10.1016/j.chaos.2020.109926
13. K. Wu, D. Dariset, Q. Wang, D. Sornette, Generalized logistic growth modeling of the COVID-19outbreak in 29 provinces in China and in the rest of the world, arXiv: Populations and Evolution (2020)
14. S.J. Taylor, B. Letham, Forecasting at scale. *Am. Stat.* **72**(1), 37–45 (2018). 10.1080/00031305.2017.1380080

A One-Dimensional CNN Model for Subject Independent Emotion Recognition Using EEG Signals



Pallavi Pandey and K. R. Seeja

Abstract EEG is a noninvasive method used to study the neural activity of the brain. This method has already proved its application in emotion recognition systems. EEG-based emotion recognition is preferred over facial-image-based emotion recognition in many cases like burned or paralyzed faces. Most of the research in this area is on subject-dependent emotion recognition where the same subject's EEG data has used for training and testing. However, in case of lack of labeled training data of a subject, there is a need for a subject-independent emotion recognition system, which could capture emotion-based EEG features common to subjects. This study aims to develop an EEG-based emotion recognition system with subject-independent approach using the publically available database DREAMER. In this research, a convolutional neural network (CNN) model, which takes raw EEG as input and classifies emotions in Valence–Arousal space is proposed for subject independent emotion recognition. The results of the proposed model have compared with the other existing studies and observed remarkable improvement on the DREAMER database. The result of the proposed CNN model shows **75.93%** accuracy for valence and **81.48%** for arousal classification on the DREAMER database for subject independent emotion classification.

Keywords Convolutional neural network · Affective computing · Valence–arousal space · One-dimensional convolution

1 Introduction

Emotions recognition comes under the field of affective computing as well as human-computer interaction (HCI). In face-to-face communications, emotions provide

P. Pandey (✉) · K. R. Seeja

Department of Computer Science and Engineering, Indira Gandhi Delhi Technical University for Women, Delhi, India

K. R. Seeja

e-mail: Seeja@igdtuw.ac.in

richer interaction and help to disambiguate meaning. Computer works as a mediator for interaction among people [1]. Therefore, emotion-enabled communication is possible with the use of computers if people are interacting from far away locations. Several theoretical aspects and applications of emotion related to HCI are covered in [2].

Emotions directly affect the personality of a human being and personality directly effects the social as well as professional life of a human being so being emotionally fit is a necessary requirement. Therefore human emotions detection is equally important as other physical disorders of the body. For better generalization of emotion recognition, there is a need to develop a subject independent emotion recognition model based on the data from various subjects instead of constructing a subject dependent model for a separate subject. This approach performs well under the assumption that there are some common features in EEG between individuals up to some extent. In real situation the things are little different as there are several challenges in EEG based emotion recognition. A major problem in recognizing emotions with EEG is that people have different emotional responses to the same stimulus. Furthermore for different stimuli of same emotion people express their emotions differently. EEG data is weak data and easily influenced by external factors, like participants' movement and environmental noise.

However, it has advantages too as EEG provides fast response and good temporal resolution. The main challenge is how to remove several artifacts so that the data quality is improved. EEG based emotion recognition has achieved good accuracy for subject dependent emotion classification but there is still scope to improve the accuracy for subject independent emotion recognition. Various studies have found that neural signatures related with different emotions do exist and there is some commonness across several sessions as well as individuals as explained in [3]. Authors used DREAMER database for emotion recognition but they worked on subject dependent approach only. The authors of the database [4] have proposed Support Vector Machine and with EEG data, they got valence accuracy 62.49% and arousal accuracy 62.17 with subject dependent methodology. Authors [5] proposed a dynamical graph CNN and with DREAMER data and subject dependent methodology as they used 17 trials data for training and one trial data for testing for each subject. They achieved 86.23% valence accuracy and 84.54% arousal accuracy. Authors [6] used hybrid deep learning combining pre-trained VGG-16 network with LSTM and got valence accuracy as 78.99% and arousal as 79.23% using subject dependent methodology. In [7], researchers have combined three databases, DEAP, DREAMER, and their own to develop a subject-independent emotion recognition system with 60 subjects and 2194 samples. They achieved accuracy is 70.26% for valence and 72.42% for arousal.

2 Materials and Methods

Emotions can be classified as a discrete set of basic emotions or as a discrete point on the Valence–Arousal space as shown in Fig. 1.

2.1 Database

This study uses publically available DREAMER database [4]. This database contains EEG recordings of 23 subjects. The Emotiv-EPOC wireless EEG headset with 14 electrodes used to record the data. Electrodes placed according to 10–20 electrode placement system as shown in Fig. 2a. Subjects watched 18 stimuli videos and provided ratings for Valence, Arousal, and dominance on 1–5 discrete scale of SAM [9] as shown in Fig. 2b. Here rating 1 is for low valence/arousal and 5 represents high valence/arousal/dominance. The Valence–Arousal model is used to get the participant's rating. The sampling rate is 128 Hz.

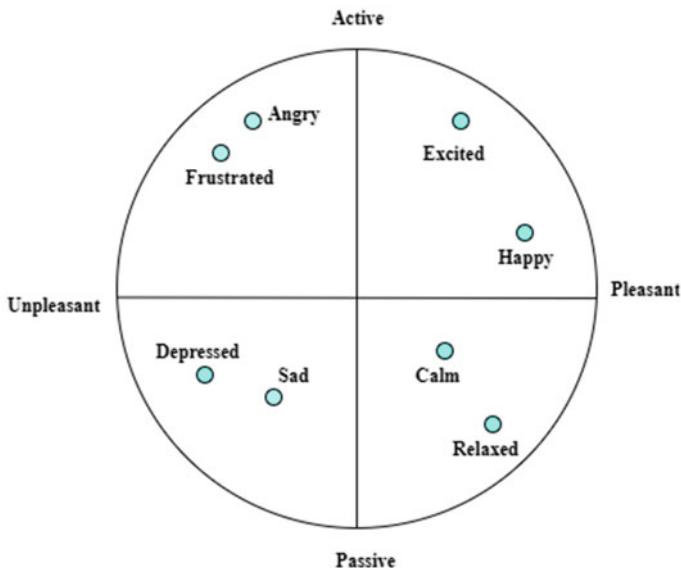


Fig. 1 Valence and Arousal model of emotions [8]

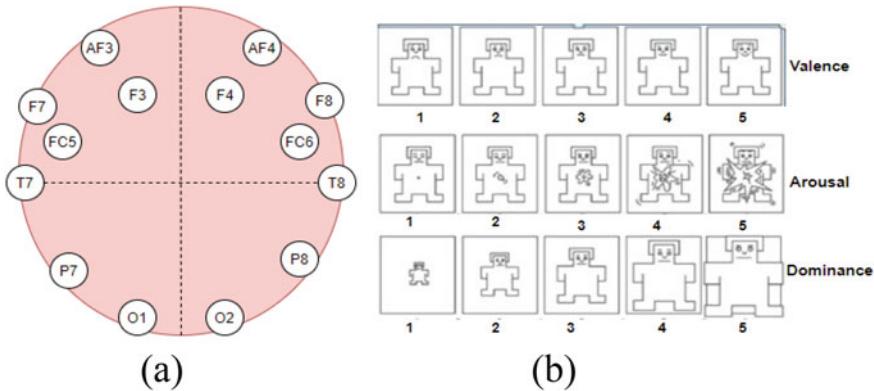


Fig. 2 Positions of 14 electrodes in emotive headset in (a) and SAM in (b)

2.2 Proposed Methodology

Emotion recognition using EEG has several phases. First, subjects (persons) watch emotion-inducing videos or images to elicit emotions while wearing an electrode cap. At that time, EEG signals are recorded using some recording software. They rate the level of emotions felt on valence–arousal scales. This scale ranges from 1 to 5 in the case of DREAMER data. The authors of the database already removed artifacts.

2.3 Proposed CNN Model

In the proposed work, one-dimensional CNN is used to recognize emotions using EEG. The architecture of CNN used for this work is shown in Fig. 3.

3 Implementation and Results

This work is implemented in MATLAB R2018b using a single GPU ‘GeForce GT 740M’ with total memory of ≈ 2 GB.

All the results are for subject-independent cases for DREAMER Database. Raw EEG from 20 subjects used to train and three subjects are used to test the CNN model separately for valence and arousal. Eighteen videos that are used to show participants to elicit emotions. Participants have rated how much arousal or valence they have felt on SAM at the discrete scale of 1–5. Three cases are considered for rating division into classes. In the first case, trials (videos) for which rating provided by subjects as ‘3’ is included in High Valence Class or High Arousal Class accordingly. For

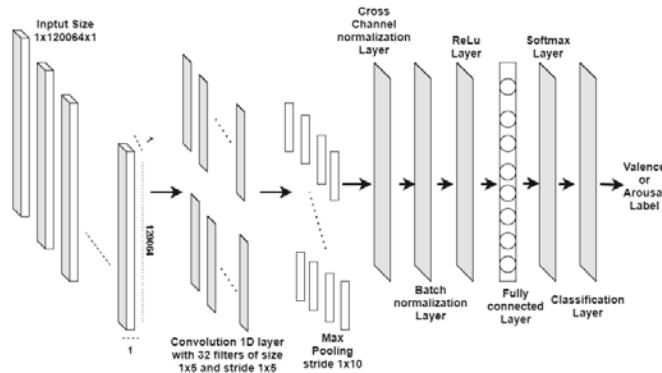


Fig. 3 Proposed CNN architecture

training data, the size of the input is 360×120064 since 20 subjects data is used for training so 20×18 , i.e., 360 inputs, and the dimension of the input is 120064 (8576×14). In the second case, trials for which rating provided by subjects as ‘3’ is included in Low Valence class and similarly in Low Arousal class. In the third case, EEG recordings with the ratings ‘1’ and ‘2’ are considered as low valence or low arousal class, similarly ‘4’ and ‘5’ are used as high valence or high arousal class and rating three data is left.

No. of convolution layers as well as filter size were the main parameters. The no. of convolution layers varying from 1 to 5 and best results were obtained at a single convolution layer. This shows one layer convolution is sufficient to capture emotion-related features from EEG corresponding to the DREAMER database. For all the three experimental cases, obtained results were publicized in Table 1 for valence and in Table 2 for arousal.

Accuracy is calculated by determining correctly identified signals divided by the total number of signals in the test set and then multiplied by 100. From the obtained results in Tables 1 and 2, it can be observed that when the EEG corresponding to those videos having rating 3 is included in low valence class or in low arousal class,

Table 1 Results of experiments for **High/Low Valence** with CNN

Experiments	Test accuracy (%)	F1-score	Training time (mm:ss)	Testing time (ms.)
Rating ‘3’ trails are included in High valence class	72.22	71.69	5:46	1.94
Rating ‘3’ trails are included in Low valence class	75.93	75.16	04:17	73
Rating ‘3’ trails are left	73.17	73.17	04:32	61

Table 2 Results of experiments for **High/Low Arousal**

Experiments	Test accuracy (%)	F1-score	Training time (mm:ss)	Testing time (ms.)
Rating ‘3’ trails are included in High Arousal class	64.81	65.15	03:00	91
Rating ‘3’ trails are included in Low Arousal class	85.19	84.69	04:45	84
Rating ‘3’ trails are left	72.22	70.01	03:25	90

Table 3 Performance comparison with DREAMER data

Article	Approach	Classifier	Accuracy (%)	
			Valence	Arousal
Arevalillo-Herráez et al. [10]	Subject Independent	Naïve Bayes	59	57
Proposed		CNN	75.93	85.19

it gives the best accuracy. Maybe the reason is that the participants were in confusion for few videos whether they are feeling low arousal/valence or high arousal/valence so they have rated the video the middle rating that is 3. But these ratings would be on the low side of the SAM scales according to the results. Performance comparison with existing study based on subject independent criteria is shown in Table 3.

4 Conclusion

In this work, a CNN model has been proposed for subject-independent emotion recognition from EEG data, where different subjects' EEG data has been used for training and testing. The result shows remarkable improvement in recognition accuracy on the DREAMER database. It has been observed that a single convolution layer is enough to find out emotion-related features from the EEG data.

References

1. S. Brave, C. Nass, Emotion in human-computer interaction, in *The human-computer interaction handbook* (CRC Press, 2007), pp. 103–118
2. R. Beale, C. Peter, The role of affect and emotion in HCI, in *Affect and emotion in human-computer interaction* (Springer, Berlin, Heidelberg, 2008), pp. 1–11
3. S. Alarcao, M.J. Fonseca, Emotions recognition using EEG signals: a survey. *IEEE Trans. Affect. Comput.* (2017)

4. S. Katsigiannis, N. Ramzan, DREAMER: a database for emotion recognition through EEG and ECG signals from wireless low-cost off-the-shelf devices. *IEEE J. Biomed. Health Inf.* **22**(1), 98–107 (2017)
5. T. Song, W. Zheng, P. Song, Z. Cui, EEG emotion recognition using dynamical graph convolutional neural networks. *IEEE Trans. Affect. Comput.* (2018)
6. S. Siddharth, T.P. Jung, T.J. Sejnowski, Utilizing deep learning towards multi-modal bio-sensing and vision-based affective computing. *IEEE Trans. Affect. Comput.* (2019)
7. H.A. Gonzalez, J. Yoo, I.A.M. Elfadel, EEG-based Emotion Detection Using Unsupervised Transfer Learning, in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (IEEE, 2019), pp. 694–697
8. S. Basu, N. Jana, A. Bag, M. Mahadevappa, J. Mukherjee, S. Kumar, R. Guha, Emotion recognition based on physiological signals using valence-arousal model, in *2015 Third International Conference on Image Information Processing (ICIIP)* (IEEE, 2015), pp. 50–55
9. M.M. Bradley, P.J. Lang, Measuring emotion: the self-assessment manikin and the semantic differential. *J. Behav. Therapy Exp. Psychiatr.* **25**(1), 49–59 (1994)
10. M. Arevalillo-Herráez, M. Cobos, S. Roger, M. García-Pineda, Combining inter-subject modeling with a subject-based data transformation to improve affect recognition from EEG signals. *Sensors* **19**(13), 2999 (2019)

Classification and Diagnosis of Alzheimer's Disease from ADNI Dataset Using RBM Classifier



Simarjeet Singh and Rekh Ram Janghel

Abstract Alzheimer's Disease (AD) is one of the chronic and irreversible sorts of disease found among grown-ups. AD causes in destroying the ability to learn, think, and remember. It is the main reason behind the death of adults as AD causes harm in the brains, i.e., presence of neurofibrillary tangles and deposition of plaques around the cerebral cortex which leads toward cognitive decline. As indicated by the previous studies around 45 million people have been affected by Alzheimer's and the rate is exceptionally quicker which may result in double the number today in the next twenty years. The main cause of AD is not known so we need to identify it at its earliest stage, which is also known as Mild Cognitive Impairment (MCI) in the medical field. The main objective of this research is to help in classifying the AD with extreme effectiveness and accuracy during MCI phase. In this research, we developed a best-in-class Restricted Boltzmann Machine (RBM) algorithm, which plays a vital role in Deep Belief Networks which is a stochastic learning method that defines a bunch of rules which is useful in feature reduction, classification, collaborative filtering, and regression methods. Additionally, the method used in this research doesn't require feature extraction and is equipped for performing Classification with the greatest effectiveness. All the experiments are performed in the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset which constitutes old age patients under two classes: Normal Class and Alzheimer Class, i.e., identifying whether Demented or Non-Demented. The methodology was divided into four sections: Normalization, Feature Reduction, Training dataset using RBM with-standing Gibbs Sampling, Contrastive Divergence, and Classification carried out in the last section. In this way, to find if our model could beat the cutting-edge models among already proposed models RBM algorithm was viably applied to the expectation of classification for Alzheimer Disease giving better results during the early phase of diagnosis. The final result shows that our model achieved an accuracy of 98.6%, specificity of 99.01%, and sensitivity of 98.22% and on differentiating the performance metrics

S. Singh (✉) · R. R. Janghel
National Institute of Technology, Raipur, India

R. R. Janghel
e-mail: rjanghel.it@nitrr.ac.in

among different techniques it shows better results. From now on, Alzheimer's Disease prediction analysis performance of our methodology recommended a unique strategy to build the predictive model among various Deep Learning strategies and Machine Learning methods. The model can perform more effectively with an increase in the size of the Alzheimer's dataset and applying optimization strategies along with RBM to reduce the computational operability and increasing effectiveness.

Keywords Alzheimer's disease · Alzheimer's disease neuroimaging initiative · Restricted Boltzmann's machine · Contrastive divergence · Principal component analysis

1 Introduction

Alzheimer's Disease (AD) is the most widely recognized type of dementia, which is a progressive brain problem that generally happens later in life. Early detection of Alzheimer's is more prominent since medicines are best whenever performed during the most prophase stage [1]. Contrasting the patient's past activities, a decrease in memory and other cognitive function is noted as an essential dementia syndrome. In 2006, the overall commonness of AD was 26.6 million, and this number is relied upon to get twice by 2046, 1.2% of the worldwide population will be influenced by Alzheimer's disease [2]. Alzheimer's disease is a sort of neurodegenerative disease of the central nervous system characterized by progressive cognitive impairment. The sixth primary driver of death with a rising population is Alzheimer's Disease and it is otherwise called late-life disease. Despite the fact that there are some clinical treatments that may briefly show constructive outcomes yet none has shown the capability of preventing deterioration [3] Alzheimer's Disease (AD) is the most well-known disease usually for individuals of 65 years age and more than 65, after some point of time the thinking capability of people steadily diminishes and arrives at a phase where it turns out to be extremely hard for them to carry on with a typical life. It is expected that 1 out of 85 individuals will be influenced by 2050 [4]. In this research, we explicitly explored utilizing a Restricted Boltzmann Machine (RBM) technique for the early identification and Classification of Alzheimer's Disease. RBM defines a set of rules which are useful namely in Classification, Regression, Feature Learning. The results determine the effectiveness of the model proposed in classifying the individual into Alzheimer's or Normal. More specifically, we proposed the study by taking the input dataset, later splitting the dataset into training dataset and testing dataset and later it is taken care by RBM model for training the model, at last Classification is analyzed utilizing various performance metrics and later best outcome appears. The model may perform more effectively if the size of the dataset is increased.

The organization of the paper is as follows: The works done in the past for early diagnosis of Alzheimer's disease are defined in the Sects. 2, 3 deals with the proposed methodology, dataset description, and techniques used in the classification of AD.

All the experimental result outcomes and discussions are defined in Sects. 4, and 5 define the conclusion of the work.

2 Literature Review

In the following years, a few motorized decisive guide approaches have been proposed, so as to lessen assessment report in inspecting the expectation of Alzheimer disease which is a repetitive and hectic duty. Different research scholars have applied Deep Learning Algorithms and different Machine Learning methodologies for the prediction of Dementia, i.e., Alzheimer's disease. A touch of the papers from the composing has been depicted underneath.

Jun Jie Ng et al. have proposed a procedure where Machine Learning techniques are used to create information on the patient's habits after some time. It was used generally to discover the area of patients around the house through the help of Estimate Bluetooth flags, and could pinpoint which room the patient was in up to an accuracy of 95% [5]. Lauge Sørensen et al. had proposed a method, where they research hippocampal texture as an MRI-based feature for distinguishing proof of Alzheimer's infection at an initial phase. Through this examination accuracy accomplished was 83% [6]. Iman Beheshtia et al. proposed a CAD-based a framework that utilizes the histogram features for the arrangement of Alzheimer's Disease (AD), where they utilized the support vector machine for order reason on cross-10-fold validations. And on this CAD framework, the accomplished accuracy was recorded as 84.07% and for order of MRI quantifies and accomplished accuracy of 97.01% for blend features of MRI measures and FAQ scores [7]. Rekh Ram Janghel proposed four different CNN architectures for the classification of Alzheimer's disease, among which ZFNet achieved an accuracy of 97.68 and 98.75% when used with 75–25 cross-validation and 90–10 cross-validation, respectively [8]. Shui-Hua Wang et al. proposed a classification of AD which was based on the eight-layer Convolutional Neural Network which attained an accuracy of 97.65% and sensitivity of 97.96% [9]. Garam Lee used a multi-modal deep learning approach for predicting Alzheimer's Disease and the method got an accuracy of 81% [10].

3 Proposed Methodology

Figure 1 represents the flow diagram of RBM on Alzheimer's Dataset, the complete flow diagram is broken into four sections: the first section deals with the pre-processing of datasets, the missing values fillings, standard scalar pre-processing, the second section deals with the dimensionality reduction that is performed using Principal Component Analysis and third phase depicts the splitting of the dataset into training and testing datasets; whereas the third section again is divided into two divisions: one is for testing and another for training. In the training section, the data

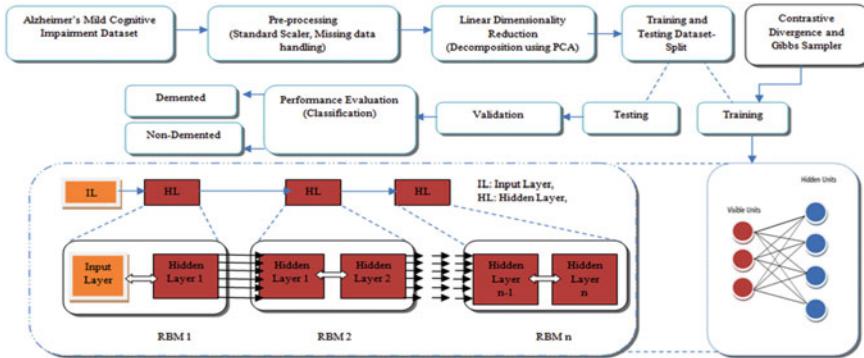


Fig. 1 Proposed Methodology of RBM for Alzheimer's Classification

is trained using the 3 layers of Restricted Boltzmann Machines and before undergoing the training phase it undergoes Contrastive Divergence and Gibbs Sampling to achieve the better performance metrics result. And the second section of the third phase undergoes testing the data, where the performance of the system is tested and different performance metrics are found out and henceforth at last stage, i.e., at the fourth stage, the classification result is carried out whether an individual is found to be demented or non-demented.

3.1 Alzheimer's Dataset

Information utilized in this paper was gained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) information base (www.loni.ucla.edu/ADNI) [11], which was dispatched in 2003 by the National Institute on Aging (NIA), the Food and Drug Administration (FDA), and the National Institute of Biomedical Imaging and Bioengineering (NIBIB) [12]. Alzheimer Disease Neurological Initiative is a worldwide research exertion that effectively bolsters the study, anatomy, and improvement in medication of Alzheimer Disease to hinder its development. The ADNI datasets contain datasets of various modalities, which can help researchers from numerous points of view for early detection of Alzheimer's malady. With its standard datasets, ADNI encourages a route for scientific researchers to lead durable exploration and offer viable information with different analysts around the globe [13]. The dataset contains a total of 3690 imaging sessions in which the patients or individuals are categorized into two different groups either in Alzheimer (Demented) or in Normal (Non-Demented). And later, in the wake of getting the dataset, we converted this image dataset into a CSV format dataset which contained 4096 features and later the data should undergo some preprocessing so that in the future when we apply a few algorithms or methodologies we will get some great outcomes. Succeeding preprocessing the dataset we separate the datasets into two different sets, first is the

training dataset with which we train our model and second is the testing dataset with which we can be certain that our model is prepared appropriately and it is and it is giving very effective results. After separating the dataset, we pick up the Restricted Boltzmann machine (RBM) algorithm and we can perform the training and testing and later classify that whether the individual is Demented or Non-Demented.

3.2 Pre-processing and Standard Scalar

Whenever any information is downloaded from the open source then there is a huge approximation that the information may contain some missing values or redundant values which may result in making our model predict to wrong results, thus there is a requirement in order to make our neural network perform better, preprocessing in the dataset is required. Either we can just remove the missing column values with an entire row or use any mathematical probabilistic approach to fill out the null spaces. In our case for the missing values, we have undergone with mean operation and filled out the null spaces or redundant spaces (we can opt for any mean, median, mode, or variance—any form of operations) so that we can let our neural network perform in a well-designed manner [14]. Whereas Standard Scaler is a machine learning preprocessing tool which standardizes the features by removing the means and scaling to the unit variance. The standardization of the dataset is very important while designing any machine learning algorithm because the model may start behaving adversely if the individual features do not more or less look like standard normally distributed data [15].

3.3 Dimensionality Reduction

Principal Component Analysis (PCA) is also termed as the linear dimensionality reduction which is an unsupervised learning algorithm which is used to extract the useful information from ADNI dataset used in this research. The dimensionality reduction is always a necessary component because there may be a possibility that a data contains a large amount of features and which may arose into a problem of Over-fitting thus PCA reduces the features and performs the ability to generalize. One of the main property of PCA is that the Principal Component (PCs) acquired are fundamentally the linear combination of the original variables, and the weight vector which is likewise the eigenvector that fulfills the property of principal of least squares [16]. In our work, the total numbers of components were equal to 1894 and observations were equal to 4096.

3.4 Restricted Boltzmann Machine

Boltzmann machines are unsupervised and non-deterministic (or stochastic) generative Deep Learning model which was proposed by Geoffrey Hinton, the model contains only two types of nodes—hidden and visible nodes. The premature optimization method utilized in ANNs depends on the Boltzmann machine [17]. At the point when the recreated toughening mechanism is applied with the discrete Hopfield organization, it turns into a Boltzmann machine. The network is designed as the vector of the conditions of the units, and the gages of the units are two valued (0 and 1) esteemed with probabilistic state progress [18].

There are no output nodes! This may seem strange but this is what gives them this non-deterministic feature. They don't have the typical 1 or 0 type output through which patterns are learned and optimized using Stochastic Gradient Descent. They learn patterns without that capability and this is what makes them so special [19].

The simple architecture and Factor Graph of RBMs can be seen in beneath diagram (Fig. 2).

RBM are two-layered Artificial Neural Networks with generative capacities. They can get comfortable with a probability apportionment over its arrangement of action, which is its type to get familiar with a probability distribution over a set of inputs of data. RBMs can be used for dimensionality decrease, classification, regression, collaborative filtering, and feature learning. Every node in the visible layer is related to every node in the hidden layer yet no two nodes in similar layers are associated with each other [20].

Working and Training of RBM

RBM is a stochastic algorithm that accepts the low-level features in order to learn the distribution of data. It comprises two layers: Hidden and Visible layers, to evaluate the output the process of RBM undergoes using this mathematical expression:

$$\text{Output } X = \text{Activation } f((w * I) + b) \quad (1)$$

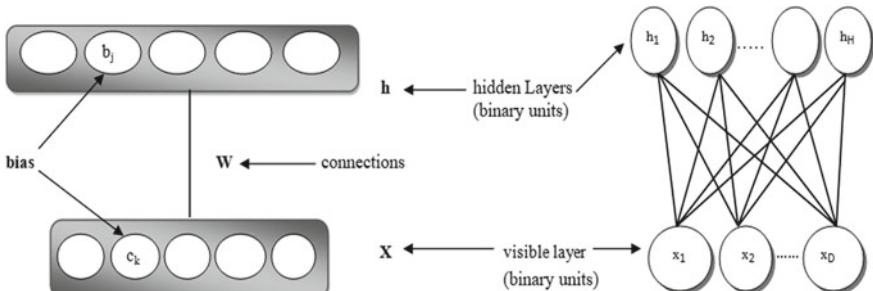


Fig. 2 Represents the architecture of RBM

where I is an input layer at a particular node, w are weights being fed at every nodes, b is the bias, and X is an output. The procedure is continued for all the other remaining layers present in the neural network. Classification has been the primary objective of RBMs [21]. We have proposed an algorithm that constitutes of three layers of RBM, the output of one layer becomes the input for the preceding next layer of RBM. In the input layer, all the weight initialization and updations are carried out later the process continues until all the hidden layers get their weights updated similar to the weight updations in Back-Propagation Networks, and at last the output is fetched which is present at the last layer of an RBM and the output is presented. Training in RBMs is a bit different than the other neural networks because it works based on the Stochastic Gradient Descent (SGD) and RBM also gets involved with two different types of methods during the neural network during the raining phase, i.e., Gibbs Sampler and Contrastive Divergence.

Contrastive Divergence (CD): CD is a Monte Carlo Markov Chain (MCMC) Gradient Descent (GD) algorithm, which perfectly suits the energy-based model Parameters or learning Products of Experts(PoE) [22]. CD is a bit tedious to solve but it gives better results than other convergence algorithms, MCMC cycle is enough to move the target data from the target distribution toward the proposed distribution and it suggests in which direction the distribution should take place to make the model better during the training phase.

Gibbs Sampler: Gibbs Sampler has become a very popular statistical tool [23]. It is a Markov Chain Monte Carlo Algorithm that is used to gather the results sequentially from the multivariate probability distribution of the information (data) [24].

4 Experimental Results and Discussions

4.1 Results

In this segment of the research, we are going to examine various performance metrics. Based on the confusion matrix and metric evaluation table, we will be performing various calculations and experimentations to evaluate the performance of the proposed classification model (Tables 1 and 2).

Table 3 represents all the performance metrics of different size-ratios with the variation of Batch size when the learning rate is 0.001 and algorithm loops up to 200 epochs.

Table 1 Confusion matrix

Actual class	Predicted class	
	Negative	Positive
Negative	[TN]	[FP]
Positive	[FN]	[TP]

Table 2 Represents the various metric names along with their mathematical expressions

Metric name	Metric mathematical representation
Accuracy	$TP + TN/Total$
Error rate	1-Accuracy or $FP + FN/Total$
Precision	$(TP/Predicted Positive) * 100$
Recall/Sensitivity	$(TP/Actual Positive) * 100$
Specificity	$(TN/TN + FP)$
F1-Score	$2 * (Sensitivity * Precision)/(Sensitivity + Precision)$

Table 3 Represents the various performance metrics of different size-ratio of training–testing with the variation of Batch Sizes when learning rate is 0.001

Batch size	Size-rate	Accuracy	Precision	Error-rate	Sensitivity/Recall	Specificity	F1-ratio
2	70–30	90.89	90.97	0.0911	90.63	91.23	90.79
4	70–30	91.64	92.10	0.0836	91.28	91.95	91.54
6	70–30	92.73	91.89	0.0727	91.32	93.34	92.63
8	70–30	91.66	91.90	0.0834	91.48	91.92	91.56
10	70–30	89.89	89.98	0.1011	89.72	90.55	89.88
16	70–30	88.96	89.15	0.1104	88.84	90.15	88.86
2	80–20	93.00	93.12	0.07	92.79	93.66	92.95
4	80–20	94.72	94.86	0.0528	94.68	94.89	94.68
6	80–20	96.53	96.67	0.0347	96.34	97.12	96.62
8	80–20	98.60	98.80	0.014	98.22	99.01	98.50
10	80–20	97.33	97.39	0.0267	97.05	97.92	97.29
16	80–20	96.20	96.42	0.038	96.02	96.89	96.14
2	90–10	92.83	92.86	0.0717	92.57	93.37	92.73
4	90–10	93.93	94.07	0.0607	93.61	94.56	93.85
6	90–10	96.22	96.45	0.0378	96.08	96.85	94.67
8	90–10	94.74	94.96	0.0526	94.69	95.83	94.68
10	90–10	96.00	96.13	0.04	95.62	96.57	95.98
16	90–10	95.80	95.88	0.042	95.49	96.96	95.70

Figure 3 represents the error-rate of different size-ratios with variation with batch size when the learning rate is 0.001.

Table 4 represents the metric accuracy with respect to the different learning rates, i.e., 0.0001, 0.001, 0.01, 0.1 with a variation on batch size with a size-rate of 80–20 (Figs. 4 and 5).

Table 5 depicts the performance metrics when different nonlinear activation function is employed with the variation in the number of epochs keeping learning rate = 0.001.

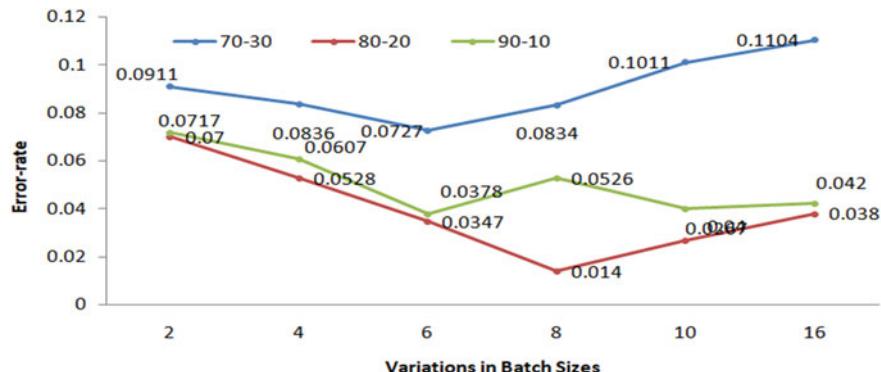


Fig. 3 Represents error-rate of different size-ratios with variation in batch sizes

Table 4 Represents the accuracy (in %) with different learning rates using 0.0001, 0.001, 0.01, 0.1 on size-rate 80–20

Batch size	0.0001	0.001	0.01	0.1
2	90.91	93.00	93.62	91.68
4	93.67	94.72	95.23	92.23
6	94.52	96.53	96.92	93.91
8	97.30	98.60	97.33	94.67
10	97.82	97.33	97.69	96.63
16	97.96	96.20	96.80	93.47

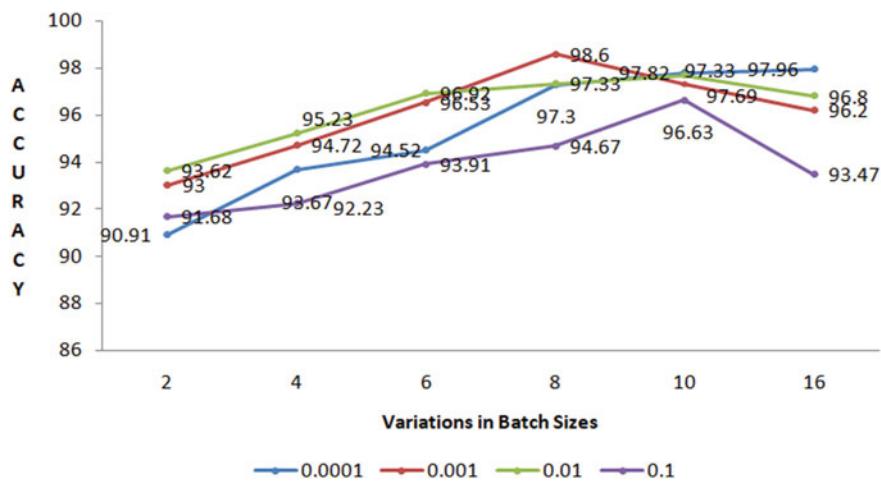


Fig. 4 Represents the metric (accuracy) using different learning rates

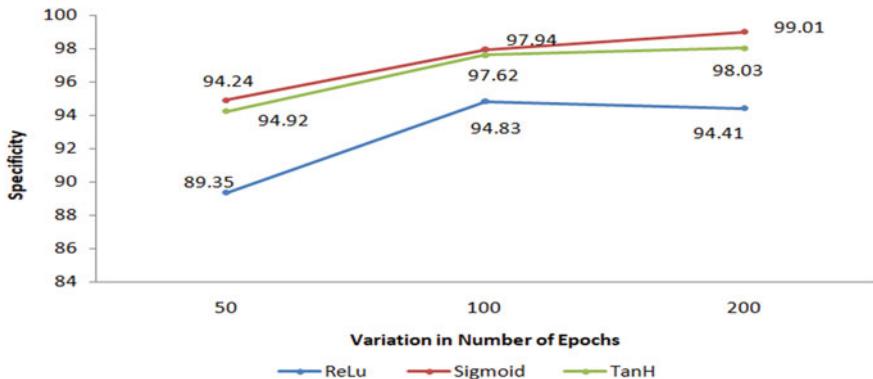


Fig. 5 Represents the specificity (in %) with the variation in the number of epochs using different nonlinear activation functions

Table 5 Represents the accuracy, sensitivity, specificity (in %) with different nonlinear activation functions using 0.001 as learning rate and size-ratio 80–20

Number of epochs	Activation function	Accuracy (%)	Sensitivity (%)	Specificity (%)
50	ReLu	88.87	88.62	89.35
50	Sigmoid	94.47	94.26	94.92
50	TanH	93.93	93.78	94.24
100	ReLu	94.41	94.22	94.83
100	Sigmoid	97.60	97.42	97.94
100	TanH	97.33	97.09	97.62
200	ReLu	93.93	93.86	94.41
200	Sigmoid	98.6	98.22	99.01
200	TanH	97.42	97.37	98.03

4.2 Discussions

The performance comparison between this work and the already proposed methods in the prediction and classification of Alzheimer's Disease is given in Table 6. From Table 6, we can observe that the proposed methodology got an accuracy of 98.6% and specificity of 99.01% which is better than some existing methods.

5 Conclusion

In this study, all the experiments are performed on the ADNI dataset, which consists of MRI Images of patients who were categorized into two classes, i.e., Alzheimer's class and Normal class. In total, the dataset had 3690 images among which 1915 were

Table 6 Represents the comparative analysis of the proposed method with already existing methods

Serial number	Authors	Method used	Accuracy (%)	Specificity (%)
1	Heung-II Suk et al. 2013 [25]	Auto-Encoders	94.90	95.42
2	Faturrahman et al. 2017 [26]	Deep Belief Network	91.76	92.96
3	Zhang et al. 2016 [27]	SVM classifier	83.7	84.25
4	Huanhuan Ji et al. 2019 [28]	ConvNets using MRI	97.65	98.19
5	Proposed	RBM	98.6	99.01

Demented and 1775 belonged to the Normal Class. On converting the image dataset into CSV dataset, the total number of features was 4096. The restricted Boltzmann Machine method proposed in this paper is used for early detection and classification of Alzheimer's disease. We used different combinations of Training and Validation datasets (70–30, 80–20, 90–10), and different combinations of parameters of the proposed strategies give effective results in terms of performance metrics. When the learning rate was fixed with 0.001, the number of epochs oscillated until 200; Sigmoid was used as an output layer activation function and then the method achieved an accuracy of 98.6%, specificity of 98.22%, and sensitivity was recorded as 99.01% which were better than the existing works in Alzheimer's disease prediction and classification. In further work, we will look at the hybrid combinations of the other Deep Learning systems and RBM methods with Swarm Intelligence strategies like Particle Swarm Optimization, Ant Colony Optimization to improve the performance metrics and the dynamics of the method.

Acknowledgements This work is supported and funded by the SEED grant project of the National Institute of Technology Raipur. The authors appreciate the support of Dr. Rekh Ram Janghel, Assistant Professor (IT Department) at NIT Raipur. We thank our sir for his constant support and guidance. Further, our sir consistently helped in every possible way and allowed us to complete the project in the right direction.

References

1. R.A. Sperling et al., Toward defining the preclinical stages of Alzheimer's disease: Recommendations from the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease. *Alzheimer's Dement.* **7**(3), 280–292 (2011). <https://doi.org/10.1016/j.jalz.2011.03.003>
2. S. Sarraf, G. Tofghi, Classification of Alzheimer's Disease Structural MRI Data by Deep Learning Convolutional Neural Networks (2016), <http://arxiv.org/abs/1607.06583>. Accessed 08 Dec 2020
3. R. Garg, R.R. Janghel, Y. Rathore, Enhancing learn ability of classification algorithms using simple data preprocessing in fMRI scans of Alzheimer's disease

4. S. Sarraf, D.D. Desouza, J.A.E. Anderson, C. Saverino, MCADNNNet: recognizing stages of cognitive impairment through efficient convolutional fMRI and MRI neural network topology models. *IEEE Access* **7**, 155584–155600 (2019). <https://doi.org/10.1109/ACCESS.2019.2949577>
5. Y. Li, J. Jie, N. Chong, W. Tan, M. Douriez, L. Thea, Machine Learning, Wearable Computing, and Alzheimer's Disease (2016), <http://www.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-91.html>. Accessed 08 Dec 2020
6. L. Sørensen et al., Early detection of Alzheimer's disease using MRI hippocampal texture. *Hum. Brain Mapp.* **37**(3), 1148–1161 (2016). <https://doi.org/10.1002/hbm.23091>
7. I. Beheshti, N. Maikusa, H. Matsuda, H. Demirel, G. Anbarjafari, Histogram-based feature extraction from individual gray matter similarity-matrix for Alzheimer's disease classification. *J. Alzheimer's Dis.* **55**(4), 1571–1582 (2017). <https://doi.org/10.3233/JAD-160850>
8. R.R. Janghel, Deep-Learning-Based Classification and Diagnosis of Alzheimer's Disease (2020), <https://www.igi-global.com/viewtitlesample.aspx?id=237939&ptid=228600&t=deep-learning-based+classification+and+diagnosis+of+alzheimer%27s+disease>. Accessed 12 Dec 2020
9. S.H. Wang, P. Phillips, Y. Sui, B. Liu, M. Yang, H. Cheng, Classification of Alzheimer's disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling. *J. Med. Syst.* **42**(5), 85 (2018). <https://doi.org/10.1007/s10916-018-0932-7>
10. G. Lee et al., Predicting Alzheimer's disease progression using multi-modal deep learning approach. *Sci. Rep.* **9**(1), 1–12 (2019). <https://doi.org/10.1038/s41598-018-37769-z>
11. M. Liu, D. Zhang, D. Shen, Hierarchical fusion of features and classifier decisions for Alzheimer's disease diagnosis. *Hum. Brain Mapp.* **35**(4), 1305–1319 (2014). <https://doi.org/10.1002/hbm.22254>
12. N. Zeng, H. Qiu, Z. Wang, W. Liu, H. Zhang, Y. Li, A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer's disease. *Neurocomputing* **320**, 195–202 (2018). <https://doi.org/10.1016/j.neucom.2018.09.001>
13. R.R. Janghel, Y.K. Rathore, Deep convolution neural network based system for early diagnosis of Alzheimer's disease. *IRBM* **1**, 1–10 (2020). <https://doi.org/10.1016/j.irbm.2020.06.006>
14. S. KumarPandey, R. RamJanghel, A survey on missing information strategies and imputation methods in healthcare, In *2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (IEEE, 2018), pp. 299–304
15. L. Buitinck et al., API design for machine learning software: experiences from the scikit-learn project, Available: <https://github.com/scikit-learn>
16. S. Wold, K. Esbensen, P. Geladi, *Decret_Du_7_Mai_1993_Fixant_Les_Modalites_D_Application_De_La_Loi_Relative_Aux_Recensements_Et_Enquetes_Statistiques.Pdf*, Chemometrics and Intelligent Laboratory Systems, vol. 2, no. 1–3, pp. 37–52, 1987, <http://files.isec.pt/DOCUMENTOS/SERVICOS/BIBLIO/Documentsdeacessoremoto/Principalcocomponentsanalysis.pdf>
17. S.K. Pandey, R.R. Janghel, Recent deep learning techniques, challenges and its applications for medical healthcare system: a review. *Neural Process. Lett.* **50**(2), 1907–1935 (2019). <https://doi.org/10.1007/s11063-018-09976-2>
18. S.N.Sivanandam, S.N. Deepa, Principles of soft computing (With CD), Google Books, [https://books.google.co.in/books?hl=en&lr=&id=CXruGgP0BTIC&oi=fnd&pg=PA1&dq=Sivanandam,+S.+N.,+%26+Deepa,+S.+N.+\(2007\).+Principles+of+soft+computing+\(with+CD\).+John+Wiley+%26+Sons.&ots=TZKZqp3Dww&sig=ujv7XWpAJ2LsHvHc2-vJe22pQzA&redir_esc=y#v=onepage&q=Sivanandam%2C S. N.%2C %26 Deepa%2C S. N. \(2007\). Principles of soft computing \(with CD\). John Wiley %26 Sons.&f=false](https://books.google.co.in/books?hl=en&lr=&id=CXruGgP0BTIC&oi=fnd&pg=PA1&dq=Sivanandam,+S.+N.,+%26+Deepa,+S.+N.+(2007).+Principles+of+soft+computing+(with+CD).+John+Wiley+%26+Sons.&ots=TZKZqp3Dww&sig=ujv7XWpAJ2LsHvHc2-vJe22pQzA&redir_esc=y#v=onepage&q=Sivanandam%2C S. N.%2C %26 Deepa%2C S. N. (2007). Principles of soft computing (with CD). John Wiley %26 Sons.&f=false). Accessed 08 Dec 2020
19. H.L. Ca, M. Mandel, R. Pascanu, Y. Bengio, B.U. Ca, Learning Algorithms for the Classification Restricted Boltzmann Machine Hugo Larochelle (2012), <http://jmlr.org/papers/v13/larochelle12a.html>. Accessed 08 Dec 2020
20. M. Jiang, Z. Pan, Z. Tang, Visual object tracking based on cross-modality Gaussian-Bernoulli deep Boltzmann machines with RGB-D sensors. *Sensors (Switzerland)* **17**(1) (2017), <https://doi.org/10.3390/s17010121>

21. M.A. Cueto, J. Morton, B. Sturmels, Geometry of the restricted Boltzmann machine, vol. 516 (2010), pp. 135–153, <https://doi.org/10.1090/conm/516/10172>
22. C. Williams, F. Agakov, Division of Informatics, University of Edinburgh Institute for Adaptive and Neural Computation An Analysis of Contrastive Divergence Learning in Gaussian by An Analysis of Contrastive Divergence Learning in Gaussian Boltzmann Machines (2002)
23. Casella, Explaining the Gibbs Sampler George, vol. 46, no. 3 (2007), pp. 167–174
24. R. Salakhutdinov, G. Hinton, Deep Boltzmann machines. *J. Mach. Learn. Res.* **5**(3), 448–455 (2009)
25. H. Il Suk, D. Shen, Deep learning-based feature representation for AD/MCI classification, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8150 LNCS, no. PART 2 (2013), pp. 583–590, 10.1007/978-3-642-40763-5_72
26. M. Ratna, W. Ito, H. Nurul, F. Moh, Structural MRI classification for Alzheimer's (2017), pp. 37–42
27. J. Zhang, Y. Gao, Y. Gao, B.C. Munsell, D. Shen, Detecting anatomical landmarks for fast Alzheimer's disease diagnosis. *IEEE Trans. Med. Imaging* **35**(12), 2524–2533 (2016). <https://doi.org/10.1109/TMI.2016.2582386>
28. H. Ji, Z. Liu, W.Q. Yan, R. Klette, Early diagnosis of Alzheimer's disease using deep learning, in *Proceedings of the 2nd International Conference on Control and Computer Vision—ICCCV 2019* (2019), pp. 87–91, <https://doi.org/10.1145/3341016.3341024>

A Comparative Study of Early Detection of Diabetes Risk by Machine Learning



Ishmeet Kaur Aubi, Swati Chauhan, and Sanjeev Kumar Prasad

Abstract One of the most essential industries around the globe is the Healthcare Industry, which consists of the manufacture of medical equipment, medicines, treatment and diagnosis of different kinds of diseases. Diabetes is a disease in which comes into play when the body loses the ability to produce insulin or when the body fails to respond to this hormone. There can be a lot of factors that can contribute to the risk of having diabetes in the early stage of life. Genetics and family history are some of the reasons that can increase the risk of diabetes. Machine learning algorithms are used to develop a mathematical model, which is based on sample data, known as “training data”, which is used to make predictions or decisions without being explicitly programmed to do so. Using these algorithms, we can predict the risk of having diabetes for an individual. The objective of this study is to give a comparative study model of the entire machine learning algorithms that have been used for early detection of the risk of diabetes with a possible maximum rate of accuracy which will, in turn, help doctors and medical practitioners to predict the risk of developing diabetes as early as possible. Its merits and demerits have also been enlisted there. In order to get the results with the maximum accuracy possible, we will do a comparative study of some machine learning algorithms like naïve Bayes, decision tree, SVM, etc. to predict the risk of diabetes for a patient. The results of this comparative study lead us to the method that is able to predict the risk for diabetes with around 94% accuracy using machine learning algorithms. This study allow us to get a clear picture of all machine learning methods used to detect the risk of having diabetes with around 94% accuracy using machine learning algorithms, which use the patient’s family history and genetics as well as their daily habits. As compared

I. K. Aubi (✉) · S. Chauhan · S. K. Prasad

School of Computing Science & Engineering, Galgotias University, NCR, Noida, India

e-mail: ishmeet_aubi.scsemse@galgotiasuniversity.edu.in

S. Chauhan

e-mail: swati_chauhan.scsemse@galgotiasuniversity.edu.in

S. K. Prasad

e-mail: sanjeevkps2002@gmail.com

to early methods for predicting risk, the machine learning algorithms are proved to be more accurate.

Keywords Diabetes · Machine learning · Health Care · Prediction

1 Introduction

In this era of technology, we came across a lot of health-related issues. Diabetes, also known as diabetes mellitus, is one of them which is quite common today that one person is suffering from diabetes out of ten. The body produces a hormone called insulin, which transfers sugar in your cells to be stored or for energy. In diabetes, the body is not able to respond properly to the insulin hormone produced or not able to produce it in enough quantity which in turn rise blood sugar level which can be quite harmful to body. There is no cure for diabetes till now but it can be treated in a manner so that the person can live a normal life.

Diabetes is also divided into two main types: Type 1 diabetes, and Type 2 Diabetes. Type 1 diabetes is an autoimmune disease, in which the immune system attacks its own body where the insulin is produced. It destroys the pancreas cells. Type 2 diabetes happens when the body builds up the resistance to insulin, which imbalances the body sugar level. Sometimes the body sugar level happens to be higher than normal due to some imbalance in hormones but not high enough to be diagnosed which rises the risk of developing diabetes in near future, that case is termed as pre-diabetes. During pregnancies, some insulin blocking hormones are produced by the placenta, which raises the sugar level in the body of the mother. This type of polygenic disease is known as Gestational Diabetes [1–3]. To sum up all, any case of diabetes can affect your nerves, eyes, kidneys, and other body organs. Therefore, early detection of diabetes is not only in debate but also a necessity to prevent it from happening in the young generation. Machine learning algorithms have proven themselves very well in doing so. Machine learning is the latest technology that has taken the IT and industries by storm with its par ability. It gives machines the ability to learn from past experiences with a given set of fixed Inputs. In this, the machine improves its performance through its past experiences. Machine Learning is mainly of three types—Supervised, Unsupervised, and Reinforcement.

In Supervised Machine Learning, we can expect a definite output on the basis of input we provide, i.e., it is similar to the situation in which a teacher teaches its student to write. In this, we are provided with a dataset whose duty is to train the model or the machine and acts as its teacher. Once the model gets trained, it is able to make a prediction or decision whenever new data is given to it. It needs labeled data to work on. It will only accomplish the task for which it will be trained. In Unsupervised Learning, the models work on unlabeled data and only a limited set of inputs is given to them. In this, the model learns through observation and improves itself in order to produce a better output. In Reinforcement Learning, the concept of hit and trial method is used. In this, the agent interacts with the environment and

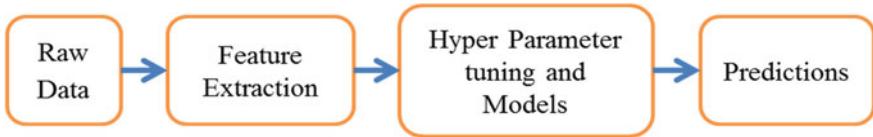


Fig. 1 The essential learning process to develop a predictive model

takes up the best outcome possible for it to produce. For each correct outcome, it will be rewarded and for each wrong outcome, it will get negative points. On the basis of rewarded points, the model trains itself. Figure 1 shows the essential learning process to develop a predictive model.

In this paper, Sect. 2 discussed the related work; Sect. 3 discusses the various methods for the prediction of Diabetic Patients and Sect. 4 discusses the review on different Machine Learning methods for early Prediction of Diabetes patients. At last conclusion of the work is given in Sect. 5.

2 Related Work

This study presents a literature review of research on the use of machine learning algorithms in the healthcare domain. The review incorporated articles presenting the knowledge about the use of the most advanced machine learning algorithms for the early prediction of one of the most common diseases that is diabetes. Diabetes is a non-contagious but deadliest disease that can lead to long-term complications and serious health problems at an early stage of life [4]. To prevent that from happening, an early Prediction of Diabetes is a must. Tejas et al. [5] proposed the three different supervised machine learning methods including SVM, Logistic regression, ANN. Survey on Type 2 Diabetes Prediction Using Machine Learning is presented in [6]. In this research, data was analyzed by different types of algorithms to avoid the risk factor of type 2 diabetes. Moreover, this paper explores the accuracy in polygenic disease, i.e., diabetes prediction in medical reports with machine learning techniques and methods. Analysis of diabetes mellitus for early prediction using optimal features selection presented in [7]. The objective of this comparative analysis is to find the optimal classifier to give the closest result comparing to clinical outcomes and get that one machine learning algorithm that will help in early detection and its future advances. The projected technique aims to focus on choosing the attributes in early detection of Diabetes Miletus using Predictive analysis. Another predictive approach is proposed in [8], this approach built the predictive models using Logistic Regression and Gradient Boosting Machine (GBM) techniques. The area below the receiver operative graphical record (AROC) was used to evaluate the discriminatory capability of those models. It used the adjusted threshold method and also the class weight method to enhance sensitivity—the proportion of diabetes patients was properly foreseen by the model. Predicting Diabetes Mellitus with Machine Learning Techniques

is presented in [9], This study used decision tree, random forest, and neural network to predict diabetes mellitus. In this paper [10], machine learning algorithms like naive Bayes, SVM, decision tree, etc., are used for the analysis of Pima India Dataset to detect the risk of developing diabetes at any stage of life based on certain factors like lifestyle, weight file(BMI), hypertension, gender, and a few others. Some similar latest papers [11–14] are used in the Machine Learning approached for the prediction of decease.

3 Methods for the Prediction of Diabetic Patients

Researchers have used varied approaches to predict the danger of Diabetic Patients in the early stage. A number of ways have been discussed in the section. Table 1 shows the accuracy of those approaches.

- (a) Decision Tree—A decision tree has found its application in Machine Learning as well. The tree includes a root internal node and leaves. Its structure sounds like a tree. It has been used in both classification and regression. It works on yes/no in which each one of them leads to a different output.
- (b) K-Means Algorithm—K-means algorithms are almost like cluster algorithms. They are a typical distance primarily based mostly on cluster methods, and space is used as a degree of likeness, i.e., the shorter distance between items specifies the more likeness.
- (c) K Nearest Neighbor Algorithm—KNN algorithms decide a parameter k, which is the nearest neighbor to that data point that needs to be classified. If in case the value of k is 5, it will look for the 5 nearest neighbors to that data point. It is seen as a result that the data point which is close to these neighbors will belong to that class only.
- (d) Logistic Regression—It is an algorithm that can be used for regression likewise as classification tasks, however, it is widely used for classification tasks. The binary response variable belongs either to one of the classes. Hence, it is used to predict categorical variables with the help of dependent variables.

Table 1 Results of algorithms

S. no.	Techniques	Accuracy (%)
1	Decision Tree	78
2	K-Means Algorithm	75
3	K Nearest Neighbor	84
4	Logistic Regression	81
5	Naïve Bayesian	79
6	Random Forest	77
7	Support Vector Machine	93

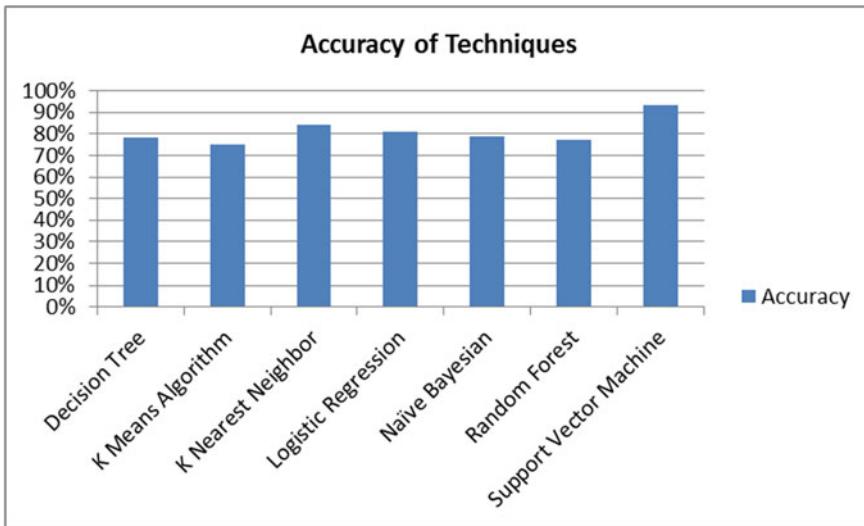


Fig. 2 Accuracy of machine learning techniques

- (e) Naïve Bayesian—Naive Bayes is a machine learning model that is used for huge volumes of data, although if you are working with data that has a large amount of data records, the best approach is Naive Bayes. It offers very good results when it involves NLP tasks such as sentimental analysis. It is an uncomplicated classification algorithm.
- (f) Random Forest—Random Forest comes from ensemble ways that are combinations of different or same type of algorithms that are used in classification tasks. Random forest is a supervised algorithm that is accustomed for both classification as well as regression tasks. In comparison to different algorithms, it is the most used algorithm when it comes to classification tasks.
- (g) Support Vector Machine—A support vector machine is a machine learning model that has the ability to generalize between two different classes if the set of labeled data is provided in the training set to the algorithm. The main function of the SVM is to check if the hyperplane has the ability to distinguish between the two classes. The accuracy of machine learning techniques is shown in Fig. 2.

4 Review on Different Machine Learning Methods for Early Prediction of Diabetes Patients

A diabetes predicting system based on a machine learning algorithm is proposed by K. Sowjanya et al. Decision tree has been used as a base for predicting diabetes in this paper. It gives results in Yes and No and whether a person is diabetic or not. Another

diabetes prediction system is proposed by Sonar et al. [15]. It uses SVM, Decision Tree, Naïve Bayes, and Artificial Neural Network to predict early diabetes in patients and compares the given results of all fours algorithms. Kumar et al. [16] use Random Forest Algorithm to predict early Diabetes. Random Forest Algorithm is used to solve real-world medical problem that is diabetes, which is one of the deadliest and chronic diseases that can happen in the early stage of life. An algorithm to predict early diabetes in an effective and affordable way is proposed by Mirshahvalad et al. [17]. The authors use the Perceptron Algorithm and Pocket Algorithm to predict the diabetes of patients. Hasan et al. [11] propose a framework that uses different machine learning algorithms and proposes results of each step from preprocessing of data to using these machine learning algorithms. Table 2 shows the proposed early-stage diabetes prediction model, used approaches, and advantages and limitations of models.

5 Conclusion

In this review, we have studied the latest diabetes prediction models and methodology is based on machine learning algorithms that can be used to detect the early risk of diabetes. We found that these algorithms are better at predicting the risk so that preventive measures can be taken in order to prevent diabetes. We saw that SVM algorithm is the most accurate in all which gives the accuracy of approx. 93%. This study can help doctors and scientists to take further steps in order to prevent this disease.

Table 2 Early stage prediction models

S. no.	Authors	Diabetes prediction model	Method used	Advantages	Limitation
1	Sowjanya et al. [18]	An ML learning based system for predicting diabetes risk using mobile devices	Decision Tree	It contains two labels yes or no, yes indicates diabetes and no indicates absence of diabetes	It must provide OFPS and OFNS scenario for the better accuracy of a diabetes prediction
2	Sonar et al. [15]	Diabetes prediction using different Machine learning approaches	Decision Tree, Naive Bayes Classifier, SVM Classifier, Artificial Neural Network	Even with unstructured and semi-structured data like text, images, and trees SVM algorithm works well. The	The drawback of the SVM algorithm is to achieve the best classification results for any given problem, several key parameters are needed to be set correctly
3	Vijiya Kumar et al. [16]	Random Forest Algorithm for the prediction of diabetes, the detection of genetic defect at its early stage	Random Forest	For applications in classification issues, it will avoid the overfitting problem, and it may be used for finding out the foremost vital options from the coaching dataset	It works on only single patient data, not on whole hospital data
4	Mirshahval et al. [17]	Diabetes Prediction Using Ensemble Perceptron Algorithm	Perceptron Algorithm, Pocket Algorithm, the proposed algorithm	Ensembles boosting algorithm with perceptron algorithm, which uses more than one weight vector for the test data classification	Execution time overload of the proposed algorithm is not considerably higher compared to the perceptron algorithm

(continued)

Table 2 (continued)

S. no.	Authors	Diabetes prediction model	Method used	Advantages	Limitation
5	Hasan, et al. [11]	Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers	Proposed framework, Evaluation metrics	The proposed ensemble model (AB+XB) yields the best performance concerning SN, FOR, and AUC by improving the XB by the margin of 2.1%, 0.8%, and 0.6%, respectively	AUC is a better modeling one that contains all the types of attributes without excluding it

References

1. V.S. Lakshmi, V. Nithya, K. Sriprya, C. Preethi, K. Logeshwari, Prediction of diabetes patient stage using ontology based machine learning system, in *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, Pondicherry, India (2019), pp. 1–4, <https://doi.org/10.1109/icscan.2019.8878831>
2. K. Driss, W. Boulila, A. Batool, J. Ahmad, A novel approach for classifying diabetes' patients based on imputation and machine learning, in *2020 International Conference on UK-China Emerging Technologies (UCET)*, Glasgow, United Kingdom (2020), pp. 1–4, <https://doi.org/10.1109/ucet51115.2020.9205378>
3. S.M. Jacob, K. Raimond, D. Kanmani, Associated Machine Learning Techniques based On Diabetes Based Predictions, in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, Madurai, India (2019), pp. 1445–1450, <https://doi.org/10.1109/iccs45141.2019.9065411>
4. F. Mercaldoa, V. Nardoneb, A. Santoneb, Diabetes mellitus affected patients classification and diagnosis through machine learning techniques. *Procedia Comput Sci* **112**, 2519–2528 (2017). Elsevier
5. S. Dewangan et al., *Int. J. Eng. Res. Appl.* **8**(1), (Part -II), 09–13 (2018). ISSN: 2248-9622
6. P.M.S. Sai, G. Anuradha, V. P. kumar, Survey on type 2 diabetes prediction using machine learning, in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India (2020), pp. 770–775, <https://doi.org/10.1109/iccmc48092.2020.iccmc-000143>
7. N. Sneha, T. Gangil, Analysis of diabetes mellitus for early prediction using optimal features selection. *J. Big Data* **6**, 13 (2019). <https://doi.org/10.1186/s40537-019-0175-6>
8. H. Lai, H. Huang, K. Keshavjee et al., Predictive models for diabetes mellitus using machine learning techniques. *BMC Endocr. Disord.* **19**, 101 (2019). <https://doi.org/10.1186/s12902-019-0436-6>
9. Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, H. Tang, *Front. Genet.* **9**, 515 (2018). Published online 2018 Nov 6. 10.3389/fgene.2018.000515; PMCID: PMC6232260
10. R. Joshi, M. Ialehegn, Analysis and prediction of diabetes diseases using machine learning algorithm: ensemble approach. *Int. Res. J. Eng. Technol. (IRJET)* **10**, 2395–0072 (2017)
11. M.K. Hasan, M.A. Alam, D. Das, E. Hossain, M. Hasan, Diabetes prediction using ensembling of different machine learning classifiers. *IEEE Access* **8**, 76516–76531 (2020). <https://doi.org/10.1109/ACCESS.2020.2989857>

12. J.A. Alzubi, A. Kumar, O.A. Alzubi, R. Manikandan, Efficient approaches for prediction of brain tumor using machine learning techniques. Indian J. Public Health Res. Dev. (2019), <https://doi.org/10.5958/0976-5506.2019.00298.5>
13. O.A. Alzubi, J.A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, M. Ramachandran, An optimal pruning algorithm of classifier ensembles: dynamic programming approach. Neural Comput. Appl. (2020), <https://doi.org/10.1007/s00521-020-04761-6>
14. D. Gupta, J.J.P.C. Rodrigues, S. Sundaram, A. Khanna, V. Korotaev, V.H.C. Albuquerque, Usability feature extraction using modified crow search algorithm: a novel approach. Neural Comput. Appl. (Springer, 2018), <https://doi.org/10.1007/s00521-018-3688-6>
15. P. Sonar, K. JayaMalini, Diabetes prediction using different machine learning approaches, in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India (2019), pp. 367–371, <https://doi.org/10.1109/iccmc.2019.8819841>
16. K. VijiyaKumar, B. Lavanya, I. Nirmala, S.S. Caroline, Random forest algorithm for the prediction of diabetes, in *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, Pondicherry, India (2019), pp. 1–5, <https://doi.org/10.1109/icscan.2019.8878802>
17. R. Mirshahvalad, N.A. Zanjani, Diabetes prediction using ensemble perceptron algorithm, in *2017 9th International Conference on Computational Intelligence and Communication Networks (CICN)*, Girne (2017), pp. 190–194, <https://doi.org/10.1109/cicn.2017.8319383>
18. K. Sowjanya, A. Singhal, C. Choudhary, MobDBTest: a machine learning based system for predicting diabetes risk using mobile devices, in *2015 IEEE International Advance Computing Conference (IACC)*, Bangalore (2015), pp. 397–402, 10.1109/IADCC.2015.7154738

Monitoring of the COVID-19 Cases by EWMA Control Chart



Pulak Kumari, Anurag Priyadarshi, Amit Kumar Gupta,
and Sanjeev Kumar Prasad

Abstract The Exponentially Weighted Moving Average (EWMA) is a measurement for observance of the method that averages the information to anticipate in perioperative outcomes to enhance the choice creating on the long run course of actions. COVID-19 an outbreak of to novel coronavirus that is determined in December 2019 and in an Asian country first case was found in Kerala. Currently, it is unfolded in the whole country as a disaster pandemic. It principally affects older people, and people with underlying clinical problems like VAS disease, diabetes, chronic metabolism disease, and cancer are a lot of in all probability to improve serious illness. The pleasant way to stop and sluggish down transmission is to monitor COVID-19 virus and the way it spreads. Defend yourself from infection by laundry your palms or the usage of sanitizer, rub it frequently and no longer touching your face. Therefore, to spot the danger of COVID-19 in future, this paper uses an EWMA management chart to examine the COVID-19 cases within the atmosphere and what is going to be impacted in the future. The EWMA control technique is a study based on statistics which is an exponentially weighted average of all prior data. It basically depends on two parameters: The first one is ' λ ' which must satisfy $0 < \lambda \leq 1$ and the second parameter is 'L' which is a set of 3 control charts. It is for a slightly small value than λ . The EWMA chart computes successive observations by computing the rational subgroup average. The EWMA accumulates information from successive readings and signals a change when a shift occurs, even if the shift is relatively very small, the EWMA chart technique detects it. We have used this EWMA control chart technique and methodology to read n detect every small change occurred in the data observed related to COVID-19 cases in India till December 21, 2020. We found that there was a rapid increase in new cases of COVID-19 patients and deaths. Later after

P. Kumari (✉) · A. Priyadarshi · A. K. Gupta · S. K. Prasad

School of Computing Science of Engineering, Galgotias University, NCR, Noida, India

e-mail: pulak_kumari.scsemca@galgotiasuniversity.edu.in

A. Priyadarshi

e-mail: anurag_priyadarshi.scsemca@galgotiasuniversity.edu.in

A. K. Gupta

e-mail: amit_gupta.scsemca@galgotiasuniversity.edu.in

the lockdown was announced by the Indian Government, the COVID-19 cases were controlled as shown in different EWMA charts.

Keywords COVID-19 · Pandemic · Global economy · EWMA · Future forecasting

1 Introduction

In this paper, we use the Exponentially Weighted Moving Average (EWMA) control chart that uses the statistics for observing the process that averages the data. The main role of the EWMA control chart is to identify the small changes from the process average with the help of past data. By the choice of the weighting factor, λ , the EWMA control procedure can be used sense even a small drift in the process. The EWMA control chart is a way of looking at variation over time. The objective of using an EWMA control chart is to detect small shifts in the process average quickly. The values of EWMA are then plotted on control charts.

The COVID-19 pandemic has generated a public health crunch, with epidemiological models predicting dire consequences, including unprecedented death rates. Due to its high contagion rate, health professionals around the world have advised maintaining hygiene and social health distancing [1].

The spread of COVID-19 can be classified [2] under three major stages:

- (a) Local epidemic: at this stage, the spreading chain of the virus among the people is tracked, and therefore the supply of infection can be found out. The cases during this stage largely relate to at intervals family or friends, or the native exposure.
- (b) Community diffusion: at this point, the source of the chain of infected persons cannot be determined. The infected cases grow through cluster transmission in the communities.
- (c) Large-scale transmission: At this stage, the virus is spreading rapidly to other regions of a country on a large scale due to the uncontrolled mobility of people.

On March 24, 2020, the government of India under Honorable Prime Minister Shri Narendra Modi ordered a countrywide lockdown of 21 days till April 14, 2020, and the lockdown was placed when the number of confirmed positive coronavirus cases in India was approximately 500. It was the First Phase of Lockdown in India. Also on March 22, a 14-h voluntary public curfew was ordered by the Government of India. The Second Phase of Lockdown was of 19 days from April 15, 2020 to May 3, 2020 which was extended on April 14 by Shri Narendra Modi with conditional relaxations in some regions where the spread of disease was minimal. On May 1, the government again extended the lockdown till May 17, 2020 which was from May 4 and that was the Third Phase of lockdown in India. The number of Corona Patients increased to 3932 from 500 in India as per the data recorded on May 14, 2020. The last phase of lockdown in India, which was from May 18, 2020 to May 31, 2020

for 14 days was extended on May 17, 2020 by The National Disaster Management Authority. The new cases observed in data analytics on May 31, 2020 were 8782.

In this paper, we represent the COVID-19 cases of major metro cities of India by using the EWMA control chart. Different EWMA control charts represent the rapid change in new cases, deceased cases, recovered and death cases of COVID-19 in India till December 21, 2020.

2 Related Work

This literature review is considered the COVID-19 future forecasting papers. In this regard, the study was proposed in Ref. [3], which used the Machine Learning models to forecast the number of upcoming patients affected by COVID-19. The study is depended on the number of newly infected cases, the number of deaths, and the number of recoveries. The study proposed in Ref. [2] is about the evaluation of COVID-19 outbreak and execute predictions using Autoregressive Integrated Moving Average (ARIMA), which is a model that explains the time series based on its own past values or data. The forecasting results in advance preparation in healthcare and to assist the government to plan the policies. This study is also based on ARIMA time series along with Eigenvalue Decomposition of Hankel Matrix (EVDHM) for nonstationary time series in Ref. [4]. It will evaluate the model by estimating the future value of daily new cases of pandemic disease COVID-19 in India, US, and Brazil. A strong literature study describing the application of Bayesian sequential and adaptive dynamic that estimates surveillance of objects that is proposed in Ref. [5]. It will help in conquering with fluctuating data of COVID-19 cases in the world. It can predict the epidemiological curve on the basis of daily recorded data. The proposed model in Ref. [6] is related to Deep Learning taken in a large number of heterogeneous features and develop complex interactions between these features. It can represent multivariate time-series and multivariate special time-series data observed during the period of the COVID-19 epidemic in the world.

3 Calculating the EWMA

To calculate the EWMA (z_i), this paper has decided on a weighting factor of 0.9. This defines how much weight is given to previous data points. It is common to use 0.2. That was the value of 0.9 used in Figs. 4, 6, 8, 10, 11, and 12. The value of z_i for sample i is estimated by Eq. 1 [7]:

$$EWMA_i = z_i = \lambda X_i + (1 - \lambda)z_{i-1} \quad (1)$$

where

$$z_i = i\text{th EWMA}$$

X_i =ith sample result

λ = the weighting factor ($0 < \lambda \leq 1$)

z_{i-1} = ($i - 1$)th EWMA

4 Methodology: Assessment of EWMA Control Charts for Observing Spread of Covid-19 Cases in India by Using EWMA

Control charts are the most often used as quality management tools and committed to monitoring and detecting shifts in the manner parameters over time. Control charts are based on the assumption that data are typically disbursed and the monitored attribute is independently distributed. This paper uses the EWMA control chart to present the COVID-19 data. Control charts are designed to observe the instability of the monitored processes. In the manage chart, the central line is located at a stage equal to the average of the monitored characteristic. In addition to this, there are two manipulate limits (upper and lower) and two warning strains (upper and lower). There are countless units of rules that can be used in monitoring methods and help to detect developments in the time series. We consider all three types of COVID-19 cases such as newly infected cases, the number of deaths, and the number of recoveries. In Table 1, data recorded of COVID-19 cases till December 21, 2020 in India of four metropolitan cities Delhi, Mumbai, Chennai, and Kolkata in four categories, i.e., confirmed, active, recovered, and deceased. Table 1 data is taken from Ref. [8].

The chart shown in Fig. 1 is representing COVID-19 cases which are confirmed, active, recovered, and deceased cases of four major metro cities: Delhi, Mumbai, Chennai, and Kolkata. As we know that in Maharashtra COVID-19 cases are increasing rapidly as shown in the chart with the orange line highest among all the four cities. Kolkata has the lowest number of COVID-19 cases. Delhi recorded fresh corona cases The highest single-day rise was in August, pushing the infection tally to over 1.62 lakh.

Total samples of COVID-19 cases are recorded with the help of EWMA in which negative cases are shown with red color and positive cases are shown with green color in Fig. 2. The data is collected for the Kaggle website [9]. These are the charts of active or negative cases by state reports.

Table 1 Covid-19 case in the year 2020

	Till December 21, 2020	Delhi	Mumbai	Chennai
Confirmed	617,808	1899,352	807,962	
Active	9255	59,469	9495	
Recovered	598,249	1789,958	786,472	
Deceased	10,304	48,801	11,992	

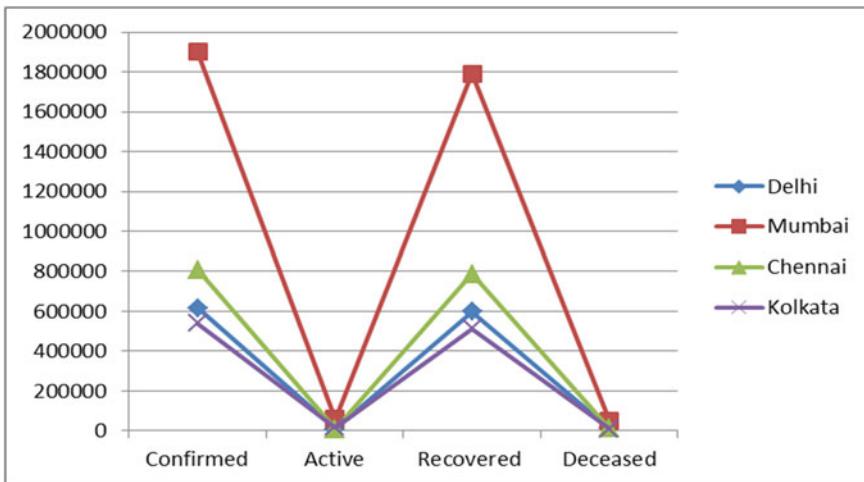


Fig. 1 Total COVID-19 confirmed cases, active cases, recovered cases, and deceased cases of major metro cities

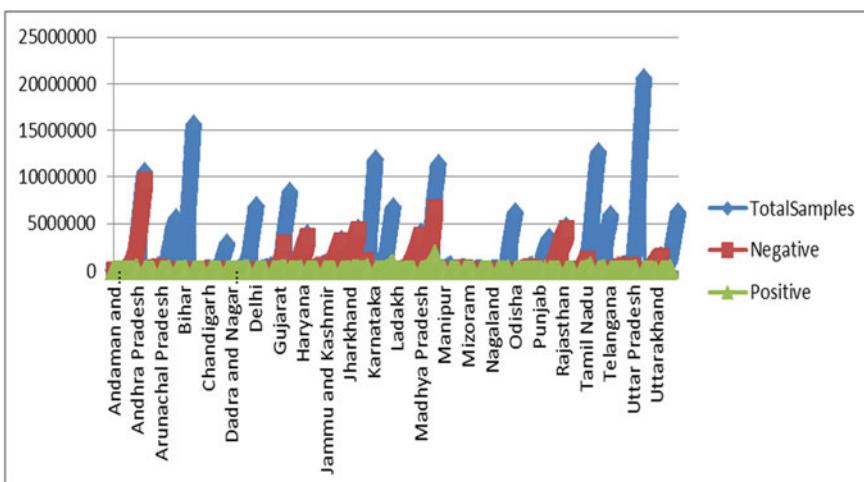


Fig. 2 Total COVID-19 cases with negative and positive cases

Figure 3 shows that till December 16, COVID-19 cases were 18,172 and it increased continuously till December 19, which was 26,834. And decreased to 19,147 till December 21. Figure 3 data is taken from the reference [8]. India recorded 10,064 fresh cases of coronavirus. The lowest case daily rise since June 11, 2020.

Figure 4 is constructed with the help of the EWMA control chart technique that are confirmed cases of COVID-19 in India till December 21, 2020. EWMA control charts shown below are representing active, recovered, deceased cases of different states and

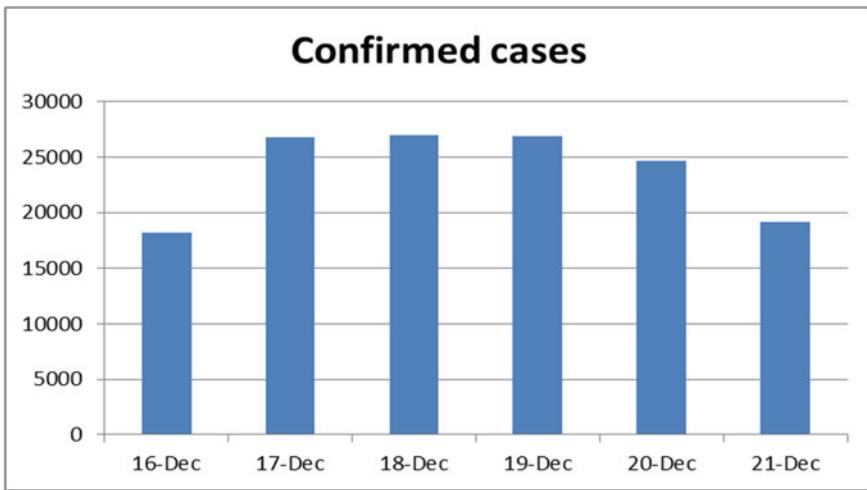


Fig. 3 Confirmed cases of India till December 21, 2020

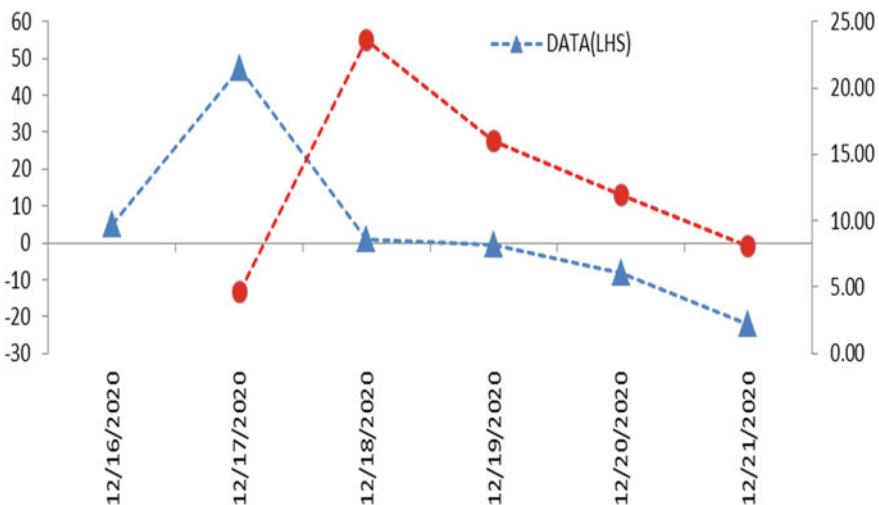


Fig. 4 EMWA control chart of confirmed cases of India till December 21, 2020

cities in India till December 21, 2020. The death toll has increased to 152,593. India's caseload tally stands at 10,582,647. Globally, nearly 96 million people have been infected by the virus. The country continues to be second-most-affected globally.

Figure 5 shows a bar chart of total active COVID-19 cases till December 21, 2020 starting from December 16, 2020 in India. The reading says that the number of cases fluctuated between these two dates [8]. The number of people who have recuperated from the disease surged to 1,01,11,294, on Tuesday pushing the national COVID-19 recovery rate to 96.49%.

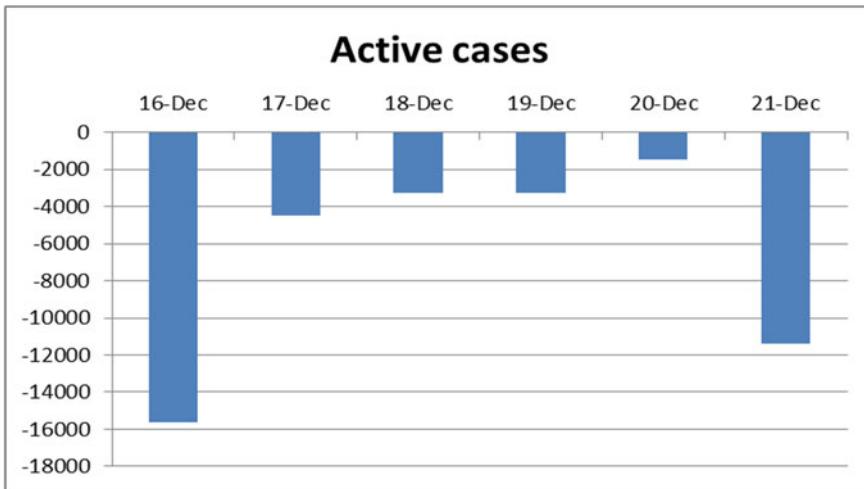


Fig. 5 Active cases in India till December 21, 2020

In Fig. 6, the blue dotted line shows the data of active COVID-19 cases and the red dotted line determines the EWMA which is numbered on the right side of the chart. Though the cases are decreased day by day the EWMA reading says that there is a high risk of increasing COVID-19 cases. The total number of coronavirus cases in the country now stands at 1,03,56,845, including 2,31,036 active cases and 99,75,958 recoveries.

The bar chart in Fig. 7 shows the recovered COVID-19 cases in India till December 21, 2020 that decreased in beginning but increased on December 21. After the Unlock

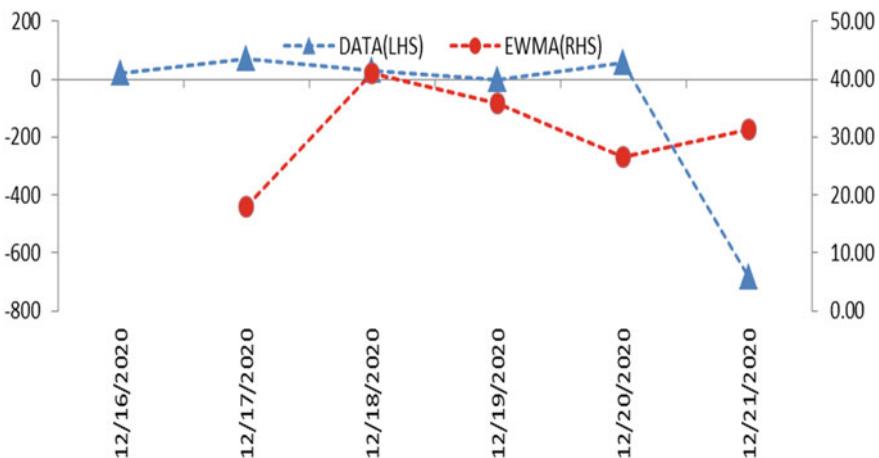


Fig. 6 EWMA control chart of active cases of India till December 21, 2020

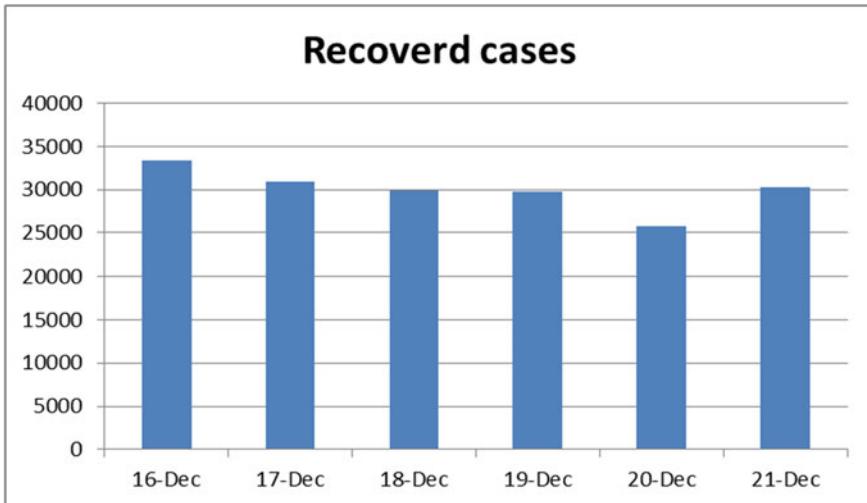


Fig. 7 Recovered cases in India till December 21, 2020

of Lockdown, the cases are increasing rapidly in some metro cities that is a matter of concern. Figure 7 data is taken from the reference [8]. From the figure, we can say that the highest recovery was on December 16 and that is around 35,000, and other dates were not following this pattern as we can see there is digression in the number of recoveries on December 20. Various studies show that disease recovers automatically after some time but causes major health problem which can lead to death, if not taken care.

The EWMA chart in Fig. 8 shows the data of the above graph in which the Exponential Weighted Moving average decreased while the data increased. As the

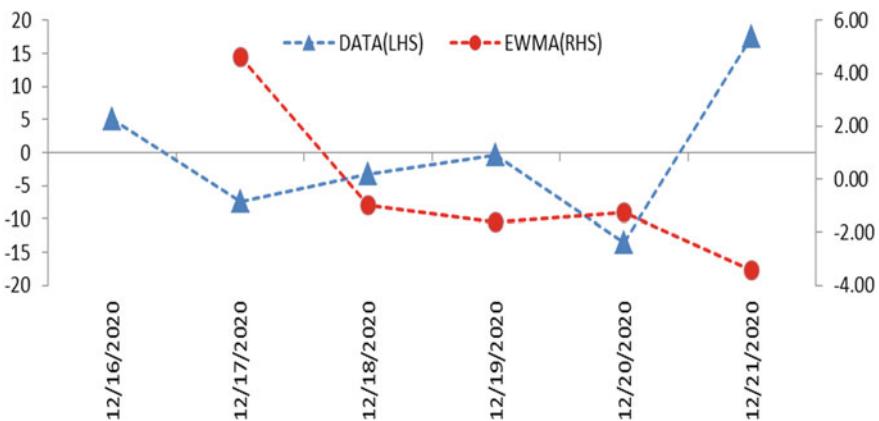


Fig. 8 EMWA control chart of Recovered cases of India till December 21, 2020

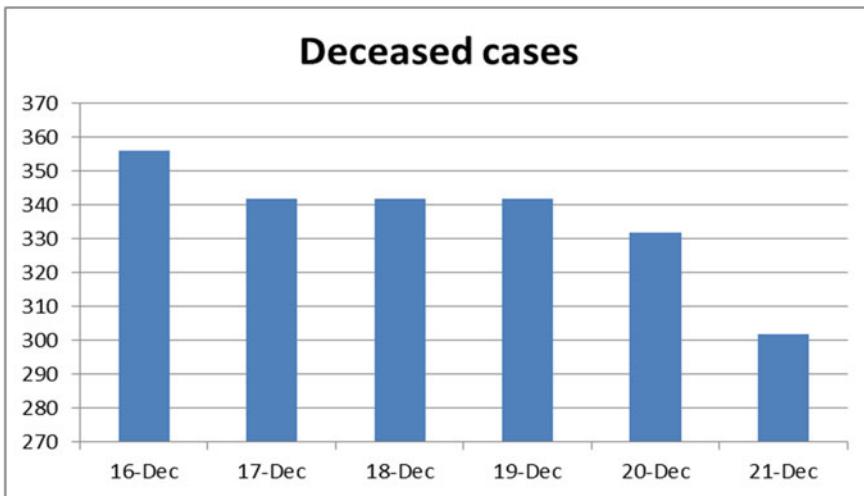


Fig. 9 Deceased cases in India till December 21, 2020

active cases increased, reading also says that the recovery is also increased as the right medications and care taken of COVID-19 positive cases. The red dotted line represents here the EWMA and on the other hand, blue dotted line showing the data of recovered cases that were discussed in the above bar chart. As we can see till December 20, both these lines moving quite in the same direction but then sudden drastic change in their direction and both ends at the opposite side.

Figure 9 shows the data of deceased cases of COVID-19 in India till December 21, 2020. The deceased cases decreased as the people started getting awareness of the precautions and proper medications under the supervision of doctors [8]. We can say that India succeeded to some extent in stopping this epidemic using lockdowns, following social distancing, etc. But still, this virus has taken the lives of many people worldwide.

The EWMA chart in Fig. 10 shows the data of the above bar graph of deceased cases of COVID-19 in India till December 21, 2020 [10]. The data shows hope of recovery from COVID-19 and decrement of death cases. Again here the red dotted line shows the EWMA, and the blue dotted lines representing the data of the deceased cases that we have seen before in the above bar chart. This gives relief as a person that we can find out here number of death rate is decreasing day by day and its average too.

The below EWMA control chart in Fig. 11 considers the Death rate of India from April 5, 2020 to December 21, 2020 due to Covid-19 cases [10]. It decreased slowly as a linear curve. As we can find out through this chart there were many ups and downs throughout the year 2020, but finally, we became able to control it somehow. This chart shows the same that how the death rate looking under control and the average as well.

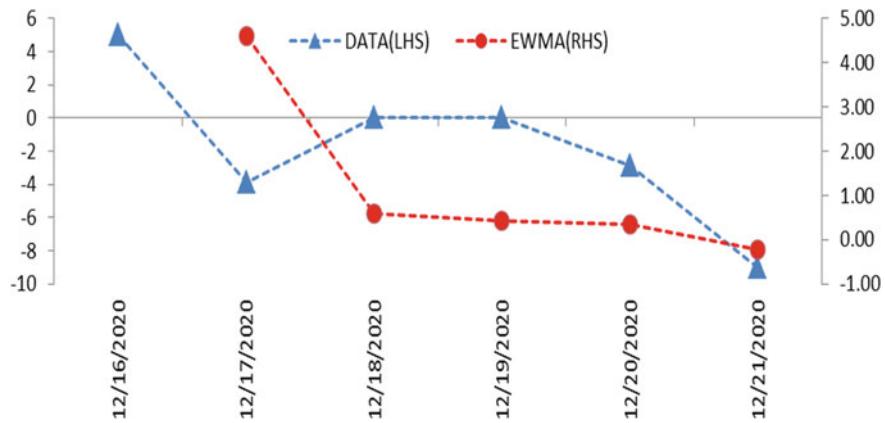


Fig. 10 EMWA control chart of Deceased cases of India till December 21, 2020

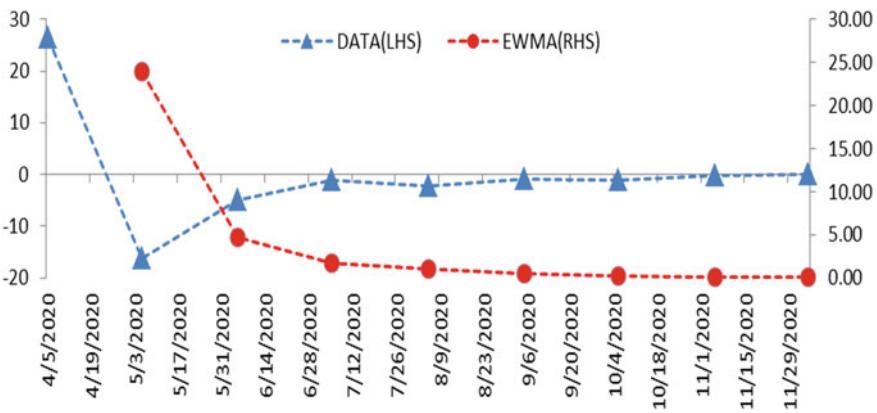


Fig. 11 EMWA control chart of death rate of India till December 21, 2020

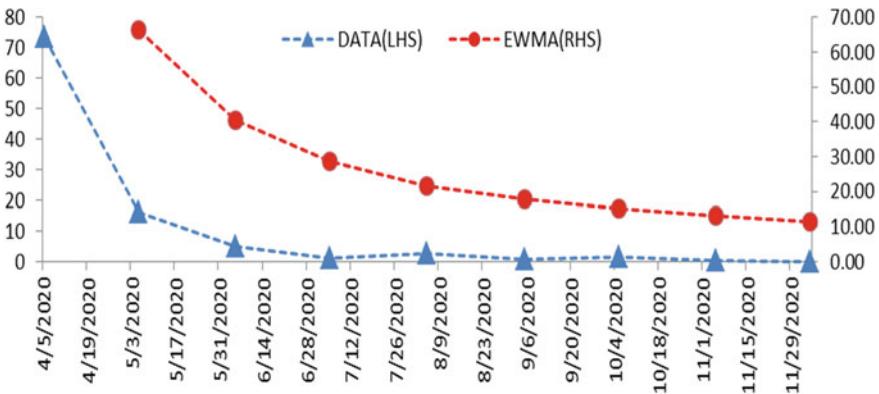


Fig. 12 EMWA control chart of recovery rate of India till December 21, 2020

The below EWMA control chart in Fig. 12 considers the recovery rate of India from April 5, 2020 to December 21, 2020 due to Covid-19 cases [10]. It decreased slowly as a linear curve.

5 Conclusion

The growth index of COVID-19 for India in first-time segment used to be 1.0588 and outcomes also indicate that the boom index is absorbed unique for exceptional states. It has been estimated that the average boom index of entire length for all the predominant states of India as nicely as an entire country.

The common boom is nonetheless extra than one indicates, in the near future the burden of the pandemic will be increasing. The upward movement suggests a warning to the authorities for the growing number of COVID-19 cases. The impact of relaxing the lockdown and insensitive behavior and wrong attitude of people let us toward severity. Some states such as Maharashtra, Andhra Pradesh, Kerala suggest the common boom index is nevertheless more than the countrywide level. Although, index values are inside the manipulation limits; however, one can now not say that India is in the desired condition. This study demonstrates a large position for statistical technique control methods to apprehend the essential factor of the time.

References

1. M.J.M. Chowdhury, M.S. Ferdous, K. Biswas, N. Chowdhury, V. Muthukumarasamy, COVID-19 contact tracing: challenges and future directions. IEEE Access, <https://doi.org/10.1109/ACCESS.2020.3036718>
2. N. Kumar, S. Susan, COVID-19 pandemic prediction using time series forecasting models, in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kharagpur, India (2020), pp. 1–7, <https://doi.org/10.1109/icccnt49239.2020.9225319>
3. F. Rustam et al., COVID-19 future forecasting using supervised machine learning models. IEEE Access **8**, 101489–101499 (2020). <https://doi.org/10.1109/ACCESS.2020.2997311>
4. R.R. Sharma, M. Kumar, S. Maheshwari, K.P. Ray, EVDHM-ARIMA-based time series forecasting model and its application for COVID-19 cases. IEEE Trans. Instrum. Measurem. **70**, 1–10, 2021, Art no. 6502210, <https://doi.org/10.1109/tim.2020.3041833>
5. D. Gaglione et al., Adaptive Bayesian learning and forecasting of epidemic evolution—data analysis of the COVID-19 outbreak. IEEE Access **8**, 175244–175264 (2020). <https://doi.org/10.1109/ACCESS.2020.3019922>
6. A. Ramchandani, C. Fan, A. Mostafavi, DeepCOVIDNet: an interpretable deep learning model for predictive surveillance of COVID-19 using heterogeneous features and their interactions. IEEE Access **8**, 159915–159930 (2020). <https://doi.org/10.1109/ACCESS.2020.3019989>
7. NIST/SEMATECH. Engineering Statistics Handbook, Published by NIST/SEMATECH (2012), 08–10
8. <https://www.covid19india.org/>, Accessed 22 Dec 22 2020
9. <https://www.kaggle.com/sudalairajkumar/covid19-in-india>. Accessed 22 December 2020
10. Johns Hopkins University Data Repository. Cssegisanddata. Accessed 22 Dec 2020, <https://github.com/CSSEGISandData>
11. <https://www.worldometers.info/coronavirus/country/india/>. Accessed 22 Dec 22

Detecting the Trend of a Product by Online Reviews Using the Supervised Machine Learning



Sangeeta Bishnoi and Rajendra Purohit

Abstract Online reviews become helpful while designing a product, it motivates consumers to post a review for the product. The fact that customer is satisfied by the product purchased by him/her can be understood by the review posted by him/her. If the review posted is positive and recommend further to purchase the product to other users, then the customer is satisfied else if negative review or comment is made and a warning experience is posted that means the customer is dissatisfied with the product purchase. MLgis at fastest growing pace in area of computer science, having numerous applications in different types of fields. Machine learning tools have capacity to self-learning patterns which has the ability to adapt and learn. As the amount of data is increasing and becoming easily available, so it's a better option to state that smart analysis of data that is often considered as a key element for progressing technology. Users frequently make errors throughout analysis, when relationships are established among multiple features. The process begins with the pre-processing techniques for feature selection which includes elimination of stop words, tokenization, stemming and lower casing. In this paper basically two techniques are used: Naïve Bayes (NB) algorithm and optimized feature selection using Naïve Bayes. The paper compares the accuracy, recall, precision and f-measure of the model of NB and optimized feature selection using NB text classification algorithms on online reviews of products to predict its trend in market. The optimized feature selection plays an important part when working with the data mining algorithms; it is capable of reducing the ambiguity of the processor as the vector space of features are reduced. The results are discussed in the detailed manner comparing the algorithms performance metrics.

S. Bishnoi (✉) · R. Purohit
Jodhpur Institute of Engineering and Technology, Jodhpur, India

R. Purohit
e-mail: rajendra.purohit@jietjodhpur.ac.in

1 Introduction

The online reviews posted by consumers depict a rich and helpful resource for social advertisement, marketing sales and questing the feedback posted by customers knowing their mood and interests in that product. A review tells us about the perception of that service he/she used online. A review can be positive or negative as per the customers experience, whenever something is purchased online then reviews are the resource which helps to give knowledgeable and useful insight of the product. The reviews are posted for any type of service used online such as airline booking, for restaurants, for shopping, for healthcare, for education, etc. [1].

Many research works are conducted in the field of text classification of reviews which include the sentiment analysis of reviews using the text classifier techniques such as naïve bayes, support vector machine, and decision trees [2]. The main problem in text classification is the high dimension of the feature space, this is often the case with text that has tens of thousands of features. Most of these features are irrelevant and not used for text classification can even reduce accuracy and a high number of features can slow down the classification process or even make some classifiers inapplicable [3].

In today's scenario a company or an organization tend toward a business service which provides the feedback from consumers. As a company/organization progresses and makes money in return it enhances the consumer satisfaction and more number of services is offered to the consumers. Mainly the online market providers read the feedback answers from customers about their products and their opinions made on it, which is larger in number as everyday thousands of reviews posted by various platforms. So it becomes a tedious and sometimes a biased job for a manufacturing company to look upon each review and make decisions on trend of product. Consequently, a technique of text mining the reviews is introduced in order to process the data in beneficial way and make a developing factor for a business to work and progress. As we know we get different types of reviews which can be negative or positive.

In this research paper the related work is discussed in Sect. 2, proposed methodology is discussed in Sects. 3, 4 is the result of the research using comparison between both techniques and their performance metrics and finally Sect. 5 discusses the conclusion of the work.

2 Related Work

The text mining of reviews and sentiments has gained popularity in research area. The hurdles that come in way of text mining of sentiments and opinions are the noisy and non-structured data on apps and website. The text mining of reviews often deals the natural language processing (NLP) method of different pre-processing techniques using the dictionary for opinion mining of text as corpus, porter stemmer

for stemming of words and any language specific dictionary [3–6]. The various research papers showed the use of NLP that tried to extract the word from sentences for stemming or removing stop words of applying n-grams.

Many researchers targets on the calculation of the word's polarity which suggests the trend of the customer's interest toward it. The trend is a cluster of words from their reviews that are extracted from sentences and comparison is made on its times of occurrences [4–6].

A research paper shows the interpretive framework and distinctive characteristics with types, business, values, and challenges of big data based on the definitional aspects. In recent years, it has been increased the BDA emphasis in e-commerce. However, the disadvantage is that it has poor explored concept, which obstructs the development of theoretical and practical. The BDA explores e-commerce by system review of drawing in the literature. This paper triggers the broader discussions about upcoming research scope and the challenges and opportunities that are on the way. At the end, this study synthesizes BDA diverse concepts that provide the deeper insights with cross cutting analysis in e-commerce market [7].

One more research conducted showed that the sentiment analysis is a very tedious and challenging task which contains machine learning and web mining. As very few work has been published on sentiment classification for Arabic languages because of the lack of managing opinions as resources were not sufficient. The major problem in Arabic languages is its complexity of phrases and words that are used by Arabic web users and the dialects used are ambiguous in nature for machine learning. So to antidote this problem, a novel approach to ensemble the text classification classifiers are adopted. The base classifiers used for text classification are Naïve Bayes, Rocchio classifier and support vector machines, and the comparison is made on two types of ensemble techniques, namely, the fixed combination and meta-classifier combination. The empirical results in the improvement of classification effectiveness in terms of accuracy by using the ensemble of classifiers [8].

Naïve Bayes classifier (NBC) is useful in evaluating the mixed datasets which consists of several missing data of two types discrete and continuous [9]. There are weaknesses of NBC like the probability function is unable to detect the accuracy of the predictions, because the probability value in NB depends on feature diversity, weight optimization and feature selection for classification [10]. Additionally the problem with missing data oftenly persists in the training and testing dataset consequentially the error value is higher. The values which are missing are replaced by other values. Commonly used methods to complete the preliminary data in NBC used are the mean and the mode methods [11, 12].

Evolutionary algorithms are developed for handling issues to decrease the processing time with high-dimensional data producing a quick result. The widely adopted evolutionary algorithm for optimized selection is genetic algorithms as the feature vector space is reduced without the information being lost. To attain an optimal feature set by making use of the training and testing datasets, heuristic information on the feature selection should be performed before the classification. The combination of genetic algorithm and Bayesian theorem is applied for estimating

the missing values in the dataset, making it a significant method for feature selection. This study developed a evolutionary GA (genetic algorithm) to optimize feature selection, with high efficiency and high accuracy rate of high-dimensional data. A poor performance is there when there are many missing feature values [13].

3 Methodology

The research paper model is implemented using the software known as “Rapid-Miner”, a platform that provides an integrated development environment for performing certain applications such as text mining, data mining, business analytics, machine learning, and predictive analytics. This tool basically applicable for commercial or business purposes, additionally it can be used for training, research, education and application development. It supports the machine learning process for data mining including all steps used such as data pre-processing, visualizing the results, optimizing and validating the models [14]. The step by step process used is explained below.

Following steps shows the stepwise process of how the methodology is implemented.

- i. Gathering the data from authentic resource.
- ii. For each review mention the trending or not trending opinion based on the rating given by the customer. If rating ≥ 3 , then mark as ‘trending’ else mark as ‘not trending’ (rating scale is from 1 to 5).
- iii. Design a process with the help of various operators used for text mining and cleaning the data.
- iv. Then in the local repository, we add the training and testing data groups.
- v. The different supervised ML techniques that are selected by us and are run for the designed process.

The output is summary of results and then they are compared considering different result metrics which are accuracy, precision, recall, and f-measure.

Pre-processing of reviews text

Pre-processing of the review text is necessary for anyone who is working with text classification algorithms. The sole aim of pre-processing is to put text reviews in such a form that are predictable and analysis can be performed on them. Steps included for text pre-processing in reduction of noisy information includes the elimination of stop words and stemming of text present in the document [15]. The major methods that are used in this paper are explained below, pre-processing mainly deals with the use of Natural Language Tool Kit (nltk) [16].

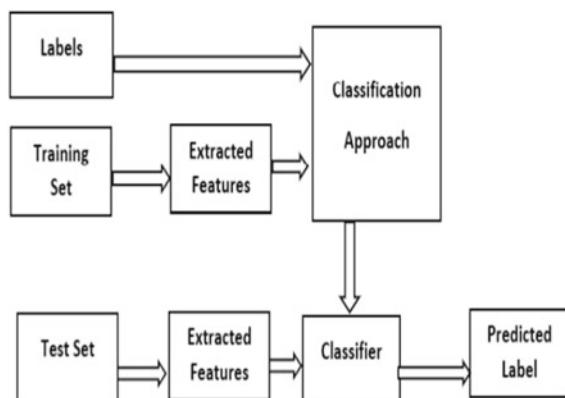
- i. *Lower casing* The basic common step for text pre-processing is lower casing the text document because while handling the text document in digital form it becomes tedious task to search for a character or a word as the development

- environment is usually case sensitive. By lower casing all the letters in the document will make out search easy by detecting the lower case letters.
- ii. *Stemming* and lemmatization are Text Normalization techniques in the field of machine learning text classification. Stemming algorithms are looked upon as rule-based. By stemming we mean the words in the document are shortened by their word stem. It can be viewed as heuristic process that removes the ends of words.
 - iii. *Removal of Stop Words* Stop words are the mostly used common words in any type of natural language. When we wish to develop model of NLP for text analysis, these stop words doesn't add worth value in the document. The list of stop words contains POS ('parts of speech') like, propositions, helping verbs, articles, or any other which is not helpful for finding out the actual meaning or the reference of a sentence.
 - iv. *Tokenization* is a technique used in text classification which breaks a series of words or sentence into chunks such as words, keywords, or symbols. This can be stated as "Tokenization is essentially splitting a phrase, sentence, paragraph, or an entire text document into smaller units, such as individual words or terms. Each of these smaller units are called tokens". When we deal with NLP tokenization in which a passage can be splitted into sentences or distinct words [17]. After extraction of the features from the training dataset it is further processed for classification modeling approach as seen in Fig. 1.

Theory of supervised learning Text Classifiers

The above-mentioned technique that is "text classification" is used to tag and assign categories to text according to its content. The reason for being known as supervised machine learning is that it consists of data which has predefined labels for classes. Classification generally deals with the labels that are pre-designed to specified tasks. The algorithm intended for text classification allocates the text to set of predefined label or class based upon its knowledge. The document or record is automatically categorized among predefined classes based upon its knowledge.

Fig. 1 Text classification technique



Selection of a Model: This step is Modeling step, in which data mining is performed on the data using the selected technique and calibrating the parameters to its optimal value which is referred as a digital representation of real-world entities. For performing modeling, there are different techniques for our selection for the same type of DM problem. Basically, model or technique is an application of algorithm used to process the data and identify it and produce the desired output.

Classification and prediction are the two steps involved in data mining. This paper uses the genetic algorithm of optimized feature selection model and proves its performance better than NB model instead of implementing the NB model alone.

- i. **Naïve Bayes** The classifier algorithm Naïve Bayes has been taken in account for the purpose of classifying the text reviews for the reason that its reliability, accuracy and efficiency. Naïve Bayes basically operates on two prototypes: (i) Bernoulli prototype—when a problem is binary classified where there are two classes, (ii) Multinomial prototype—when a problem is related to more than two classes. Taking in account how the document has classified the classes and distributed it, the calculation of “posterior probabilities” for every label in the record is performed. It disregards the original context and term’s position and on the basis of tf-idf the probabilities are calculated. The variables and its limitations for every document ought to be eventually approximated to get full benefit out of the probability of the training data features which are created by the record. The probabilities outcome is further utilized in estimating test features which belongs to that class. The need of NB classifier is to increase the accuracy with large database also, the idea behind this model is that it combines the probability of the words and classes to calculate the overall probability of the class in the mentioned document [15, 18].
- ii. **Optimized feature selection (Genetic Algorithm)** The algorithm used with NB classifier to improve its accuracy is the Genetic Algorithm (GA), which is considered as one of the optimizing algorithms. The GA is hypothetically based on the idea of natural genetic selection which in result produces better performance when dealing with optimization problems and machine learning [13]. This type of algorithm works on the principle of natural selection and natural genetics. The researcher mentioned in his paper about GA model as, “Genetic algorithms begin with a set of initial solutions (individuals) called populations. One very important thing is that one individual state one solution. The initial population will evolve into a new population through a series of iterations (generations). At the end of the iteration, the genetic algorithm returns one of the best members of the population as a solution to the problem”. From this conclusion it is concluded that the rate of success of GA model depends on two factors: selective pressure and population diversity. As the selective pressure increases, the number of chromosomes increases which are imitated from pre-generation. In contrary to this statement, it can be concluded that as population diversity increases there is decrease in proportion of inherited chromosomes which leads to loss of evolution as per offspring [17].

Evaluating and Validating the Results obtained: This is final step used for evaluating the tested data against the trained data and comparing it to the whole setup model using NB and optimized feature selection algorithm. The model which is indicating higher value of accuracy that model is considered as best in performance.

- *Accuracy value:* This type of metric is used in the evaluation of the classification algorithm which measures the ratio of the correct predictions of the data over the total number of predictions. Mathematically accuracy is represented as

$$\text{Accuracy} = \frac{\text{Frequency of Correct Predictions}}{\text{Total Frequency of Predictions}} \quad (1)$$

To be more precise in terms of binary classification, the accuracy metric is calculated for the sum of true negatives (TN) and true positives (TP), as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

where false positive and false negative are designated as FP and FN, respectively.

- *Recall:* This type of metric is defined as to measure the fraction of positive patterns/relevant features which are classified or extracted correctly. The formula can be stated as:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- *Precision:* This type of metric is used for the measurement of the positive patterns of correctly predicted tag/class to the total number of predicted patterns in a positive class. The higher the precision score sets for accurate result which takes up all significant data returning back the finest results. The formula for computing precision is:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4)$$

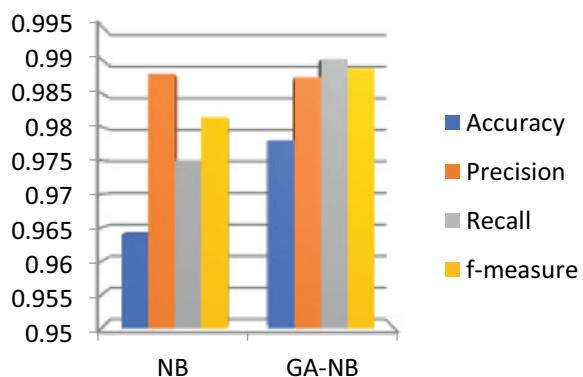
- *F-Measure:* This type of metric is usually helpful when there is some sort of balance between precision and recall. It can be defined as the harmonic mean ratio of the precision and recall. The formula is as follows:

$$F - \text{Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Table 1 Result summary of performance metrics

Metrics	NB	GA-NB
Accuracy	0.964	0.978
Precision	0.9879	0.9874
Recall	0.975	0.9901
f-measure	0.9814	0.9887

Fig. 2 A graph showing the several metrics comparison for both classifier methods—electronic products reviews dataset



4 Experimental Results

For the implementation of NB algorithm, we considered 2000 data reviews of electronic products for study. The ratio of split validation of training and testing datasets taken is 70:30, which means 70% of data is taken for training set and rest 30% data reviews are used by testing set and hence evaluates the model.

The optimized values in feature selection GA result taken are population size as 50, the number of generations as 30, p crossover as 0.8 and p mutation as 0.08 [18] (Table 1 and Fig. 2).

5 Conclusion

The optimized feature selection evolutionary algorithm combined with NB performs better as compared to NB algorithm having heterogeneous set of values. The optimized feature selection evolutionary algorithm produces an effective and less feature set resulting in optimal performance. By this way a manufacturer is able to easily justify that the product is trending in the market or not. The higher value for all performance metrics determines that the certain kind of product is in trend having all the missing values eliminated.

The research work is extended further to mine customer requirements getting the designer's concern in mind. It can be implemented further to obtain the helpfulness

score of the product as the review is helpful or not, and bringing the maximum useful information and dicing the customer requirements [19]. This research can be applied to various fields of military, manufacturer companies, traveling sites, etc. giving better understanding about the trend of certain thing in the market by evaluating the reviews of the customers posted by them and further making improvements to enhance the value of the product or idea.

References

1. S. Wu, R.D. Chiang, Z.H. Ji, Development of a Chinese opinion mining system for application to Internet online forum. *The Journal of Supercomputing*, Springer US (2016)
2. Z. Li, L. Liu, C. Li, Analysis of customer satisfaction from chinese reviews using opinion mining, in *Proceeding of the 6th IEEE International Conference on Software Engineering and Service Science(ICSESS)* (2015), pp. 95–99
3. S. Atia, K. Shaalan, Increasing the accuracy of opinion mining in Arabic, in *Proceeding of the 1st International conference on Arabic computing linguistics* (2015), pp. 106–113
4. T. Chumwattana, Using sentiment analysis technique for analyzing Thai customer satisfaction from social media, in *Proceeding of the 5th International Conference on Computing and Informatics* (2015), pp. 659–664
5. S. Ahmed, A. Danti, A novel Approach for sentimental analysis and opinion mining based on senti wordnet using web data, in *Proceeding of International Conference on Trends in Automation, Communications and Computing Technology* (2015), pp. 1–5; R.K. Bakshi, N. Kaur, R. Kaur, G. Kaur, Opinion mining and sentiment analysis, in *Proceeding of the 3rd International Conference on Computing for Sustainable Global Development* (2016), pp. 452–455
6. L. Lin, I. Li, R. Zhang, W. Yu, C. Sun, Opinion mining and sentiment analysis in social networks: a retweeting structure-aware approach, in *Proceeding of the 7th International Conference on Utility and Cloud Computing* (2014), pp. 890–895
7. R.Y. Lau, J.L. Zhao, G. Chen, X. Guo, Big Data commerce. *Inf. Manag.* **53**, 929–933 (2016)
8. N. Omar, M. Albared, Ensemble of Classification algorithms for subjectivity and sentiment analysis of arabic customers' reviews. *Int. J. Adv. Comput. Technol. (IJACT)* **5**(14) (2013)
9. C.C. Hsu, Y.P. Huang, K.W. Chang, Extended Naive Bayes classifier for mixed data. *Expert Syst. Appl.* **35**(3), 1080–1083 (2008); O. Addin, S.M. Sapuan, E. Mahdi, M. Othman, A Naive-Bayes classifier for damage
10. Received: November 7, 2019. Revised: December 12, 2019. 343
11. International Journal of Intelligent Engineering and Systems, vol. 13, No. 1 (2020), <https://doi.org/10.22266/ijies2020.0229.31>
12. Detection in Engineering Materials. *Mater. Design* **39**(12), 2379–2386 (2007); P. Domingos, M. Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss. *Mach. Learn.* **29**, 103–130 (1997); U.N. Dulhare, Prediction system for heart disease using Naive Bayes and particle swarm optimization. *Biomed. Res.* **29**(12), 2646–2649 (2018)
13. W. Shahzad, Q. Rehman, E. Ahmed, Missing data imputation using genetic algorithm for supervised learning. *Int. J. Adv. Comput. Sci. Appl.* **8**(3), (2017)
14. <https://en.wikipedia.org/wiki/RapidMiner>
15. P. Mitra, C.A. Murthy, S.K. Pal, Unsupervised feature selection using feature similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(3), 301–312 (2002)
16. H. Liu, H. Motoda, R. Setiono, Z. Zhao, Feature selection: an ever evolving frontier in data mining. *J. Mach. Learn. Res. JMLR* **10**, 4–13 (2010)
17. M.N.S. Zainudin, M.N. Sulaiman, N. Mustapha, T. Perumal, A. Shahrel, A. Nazri, R. Mohamed, S.A. Manaf, Feature selection optimization using hybrid relief-f with self adaptive differential evolution. *Int. J. Intell. Eng. Syst.* **10**(2), 21–29 (2017)

18. D.A. Muthia, D.A. Putri, Implementation of text mining in predicting consumer interest on digital camera products, in *The 6th International Conference on Cyber and IT Service Management (CITSM 2018)* (2018)
19. V. Bachu, J. Anuradha, A review of feature selection and its methods. *Cybern. Inf. Technol.* **19**(1), 3–26 (2019)
20. M. Zhao, C. Fu, L. Ji, K. Tang, M. Zhou, Feature selection and parameter optimization for support vector machines: a new approach based on genetic algorithm with feature chromosomes. *Expert Syst. Appl.* **38**(5), 5197–5204 (2011)

DiabeDetect: A Novel Decision Tree-Based Approach for Early Prognosis of Diabetes



Muhammad Usama Islam , Md. Mobarak Hossain, Iqbal Hossain, and Mohammad Abul Kashem

Abstract Diabetes is one of the most serious threats for human throughout the world and currently this is one of the most threatening cases in low and low-middle-income countries. However, the early prognosis can significantly reduce the threat to a minimal level through proper care-coordination schemes. This paper aims at finding the likelihood probability of diabetes using different machine learning techniques along with a novel approach aided by Decision Tree to improve the performance. Diabetes patients data were collected from the early stage diabetes risk prediction dataset of the UCI machine learning Repository. This dataset is consisted of 17 attributes in which we applied Logistic Regression, Naive Bayes, variants of Support Vector Machine (SVM), and our novel decision tree aided approach and calculated their prediction accuracy. An efficient exploratory data analysis coupled with the attribute correlation method helped us to develop the novel approach that improved the accuracy by deducting some lower ranked misguided attributes. After a careful selection of upper ranked attributes, we found a much-improved accuracy rate of 92% for our Novel approach that outperforms the predecessor approaches.

Keywords Diabetes · Diabetes prognosis · Machine learning

M. Usama Islam (✉)
UL Lafayette, Lafayette, LA, USA
e-mail: usamaislam@iut-dhaka.edu

Md. M. Hossain
Dhaka University of Engineering and Technology, Gazipur, Bangladesh

I. Hossain
ULKA Games Limited, Dhaka, Bangladesh
e-mail: iqbalhossain@iut-dhaka.edu

M. Abul Kashem
Dhaka University of Engineering and Technology, Gazipur, Bangladesh
e-mail: drkashemll@duet.ac.bd

1 Introduction

Diabetes is one of the most problematic diseases that has hampered the daily lives of the people. It has been estimated that by 2040 the people affected by diabetes would reach a level of 642 million of which the maximum of the population is attributed to persons from low and middle income countries [18]. This disease is harmful not only because it has the ability to impair cognition, brain function [22] but also the severity of vulnerability has doubled during COVID pandemic due to its attack to jeopardize the immune system [7]. While there remains various types of Diabetes with variants of severity on immune and cognition system [25] but it can be summarized to a unified point that Diabetes is harmful for human body [11].

Performing a prognosis of diabetes is a specialized task usually performed by certified doctors because a variety of conditions are being diagnosed to provide a diagnosis of diabetes. In this task, we investigate the data generated in Doctor-Patient session curated to a form of dataset by Islam and his colleagues [11] and available at UCI machine learning repository.¹ However, before we start to curate our novel approach, we discuss in the next section the related literature works and motivation from those related works that prompted our research.

2 Literature Review

Data mining, machine learning, big data are a few of many buzzwords that have created a spectrum change in the world after the data explosion [14, 28]. In healthcare sector, the data analysis gained substantial momentum through the technological revolution of health data records with Electronic health records (EHR) [3, 15, 20, 21]. With the advent of data in healthcare, machine learning and big data paved its way into the industry by introducing and generating insights from data [1, 9, 10]. Diabetes is a disease which has been studied to a range of wide spectrum all around the world [19, 27]. With the advent of big data in healthcare, Diabetes research has gained substantial attraction in machine learning research community of which PIMA dataset remains a popular one [5, 17]. However, a wide range of research works on detection, prognosis, diagnosis, and analysis of diabetes with care-coordination options has been carried out by the research community from which Farzana Anowar and her team has curated a comprehensive review on applications that are used by diabetes patients for self care-coordination [4].

Karatsiolis et al. [13] have taken the aid of support vector machine (SVM) for medical diagnosis purposes of diabetes. Similar works concerning classification have been performed in [6] with the major difference being the use of a new k-nearest neighbor approach which is class-wise in nature. The use of Fuzzy logic has been explored by Vaishalli and his team [26] as well as Lekkas and his colleagues [16]. Deep learning as a viable tool for prognosis of diabetes has been explored in [24].

¹ <https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes+risk+prediction+dataset>.

Similar research work but for resource constrained devices such as mobile devices has been explored by Karan and his team [12] Islam et al. [11] have curated a new dataset and performed some data mining techniques to analyze the likelihood of diabetes in their research work. Alpan and his team also has explored various data mining approach through the utilization of WEKA to classify diabetes [2].

The aforementioned research works have motivated us to perform our analytical approach in prognosis of diabetes. The remainder of the thesis is as follows; Section 3 shall contain the materials and methods of our proposed methodology, Sect. 4 has the result analysis and discussion through evaluation, and finally, Sect. 5 contains the discussion of future works and concluding remarks of our research work.

3 Materials and Methods

3.1 Dataset

The dataset that we will use in our experiment has been curated by Islam and his team [11] and available at UCI machine learning repository. The dataset has 520 entries with 17 attributes of which “Class” column is the output column that has 200 negative samples of diabetes patients and 320 diabetes positive patients. A sample of the dataset can be visualized in Fig. 1.

3.2 Dataset Preparation

The diabetes dataset has missing values as well as values of non-numeral nature that can be label encoded for our experimental purpose. For the preparation of our final dataset, we perform averaged action for missing values followed by label encoding

	Gender	Polyuria	Polydipsia	sudden weight loss	weakness	Polyphagia	Genital thrush	visual blurring	Itching	Irritability	delayed healing	partial paresis	muscle stiffness	Alopecia	Obesity
0	Male	No	Yes	No	Yes	No	No	No	Yes	No	Yes	No	Yes	Yes	Yes
1	Male	No	No	No	Yes	No	No	Yes	No	No	No	Yes	No	Yes	No
2	Male	Yes	No	No	Yes	Yes	No	No	Yes	No	Yes	No	Yes	Yes	No
3	Male	No	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes	No	No	No	No
4	Male	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
...
515	Female	Yes	Yes	Yes	No	Yes	No	No	Yes	No	Yes	Yes	No	No	No
516	Female	Yes	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	Yes	No	No	No
517	Female	Yes	Yes	Yes	Yes	Yes	No	Yes	No	No	No	Yes	Yes	No	Yes
518	Female	No	No	No	Yes	No	No	Yes	Yes	No	Yes	No	No	Yes	No
519	Male	No	No	No	No	No	No	No	No	No	No	No	No	No	No

520 rows × 15 columns

Fig. 1 Likelihood estimation of diabetes dataset

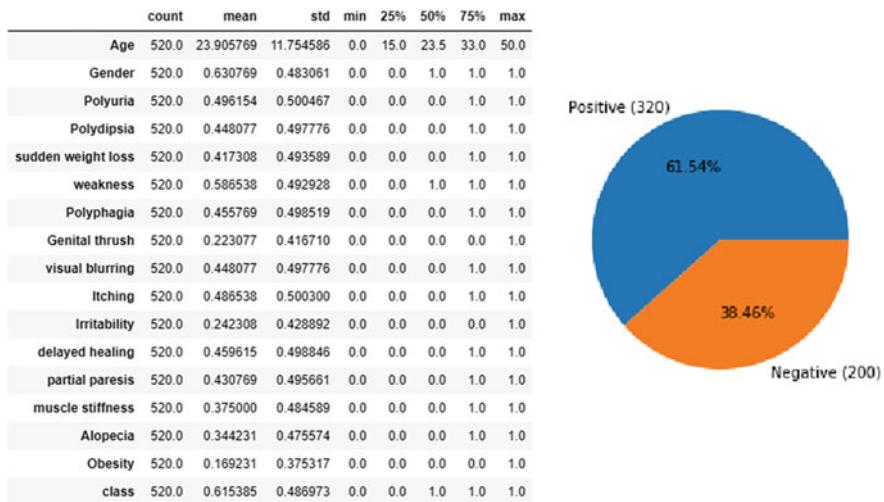


Fig. 2 Exploratory data analysis of dataset

and through an exploratory analysis of feature description by calculating the mean, median, and standard deviation as well as the correlation of features we curate our final dataset. The description of data that has the calculation of mean, median, standard deviation, etc. can be visualized in Fig. 2 and the correlation of data can be visualized in Fig. 3. From the exploratory analysis of the dataset we curate our final dataset with 16 features.

3.3 Theory of Decision Tree Approach

The decision tree works on the prediction of target variable based on the information fetched through the feature variables. The decision tree measures a probability distribution of conformity where a certain class belongs. The procedure includes split of a source set into subsets with repetition often referred to as recursive partitioning [8].

3.4 Metrics of Decision Tree

As we know that decision tree splits a source sets into subsets on the basis of a probability distribution. But, how exactly the decision is taken for choosing the best possible values out of several probabilities, metrics are often implemented. In our research work, we have taken Entropy, Gini impurity, and information gain as our

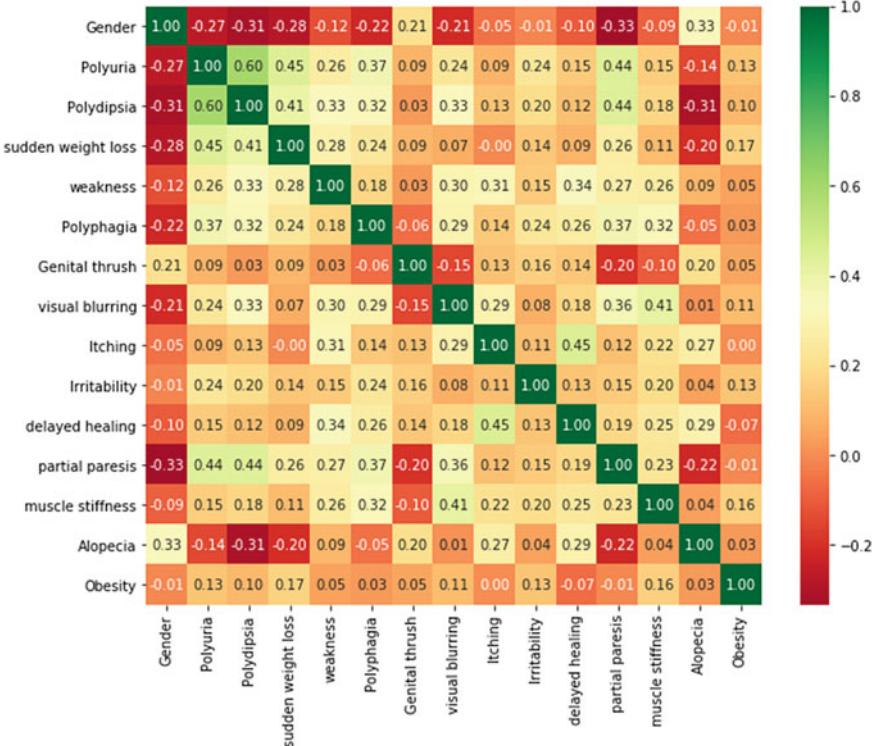


Fig. 3 Data correlation of dataset

metrics [23]. The formula for entropy with its variation of gini impurity is equated in Eq. 1.

$$E(D) = \sum_{i=1}^c -p_i * \log_2(p_i) \quad (1)$$

In our research work, apart from entropy, we shall interchangeably use gini impurity as well because it does not use logarithmic results thus rationale being less complication in calculation. The formula for gini impurity is provided in Eq. 2.

$$E_{gini}(D) = \sum_{i=1}^c -p_i * (1 - (p_i)). \quad (2)$$

As entropy measures the impurity, the effectiveness of attribute is measured with information gain. The equation of information gain is provided in Eq. 3.

$$G(D, A) = E_{fin}(D) - \sum_{v \in val(A)} \frac{D_v}{D} * E_{fin}(D_v). \quad (3)$$

4 Evaluation and Result Analysis

4.1 Experimental Evaluation

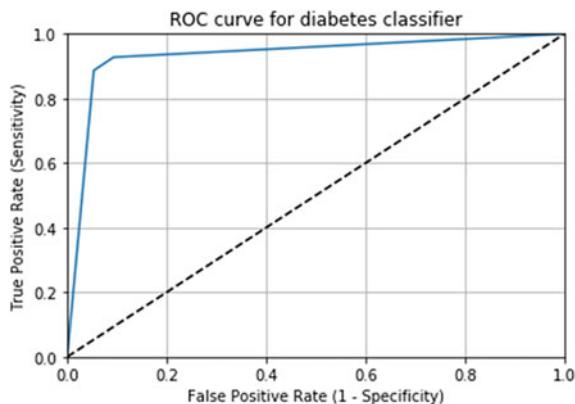
Our approach, DiabeDetect is trained on 67% of the data and tested on 33% of the data to a total of 172 instances. Our approach performed with a staggering accuracy of 92% and Area under the Curve (AUC) being 93.4%. The full classification report can be observed at Table 1. We have plotted the true positive rate (Sensitivity) in function of the false positive rate thus generating a Receiver Operating Characteristic (ROC) curve which can be visualized in Fig. 4. The confusion matrix of our experiment can be visualized in Fig. 5.

Furthermore, we have curated a table to understand the ultimate contribution of features in determining the likelihood of diabetes. Table 2 provides with the insights into subsection of importance such as $importance \geq 0.024$ and $importance \leq 0.024$ to understand the contribution of the features.

Table 1 Classification report of DiabeDetect

Decision	Precision	Recall	F1-score	Support
Diabetes (Neg)	0.91	0.91	0.91	75
Diabetes (Pos)	0.93	0.93	0.91	97
Accuracy			0.92	172
Macro Average	0.92	0.92	0.92	172
Weighted Average	0.92	0.92	0.92	172

Fig. 4 Receiver Operating Characteristic (ROC) curve of DiabeDetect



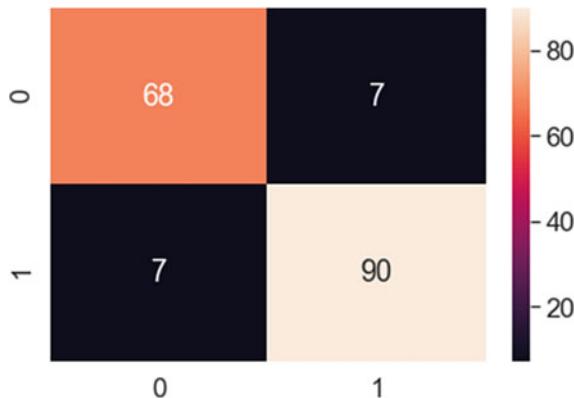


Fig. 5 Confusion matrix of Diabedetect

Table 2 Feature importance of DiabeDetect

Feature name	Importance(≥ 0.024)	Feature name	Importance(≤ 0.024)
Polyuria	0.471631	Delayed healing	0.011027
Alopecia	0.129868	Weakness	0.007025
Gender	0.104787	Sudden weight loss	0.006721
Polydipsia	0.096896	Itching	0.006486
Genital thrush	0.070527	Polyphagia	0.006301
Muscle stiffness	0.028950	Partial paresis	0.005251
Visual blurring	0.027179	Obesity	0.006301
Irritability	0.024704		

4.2 Comparative Analysis and Discussion

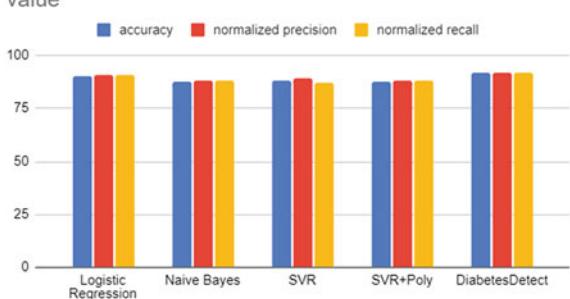
We have compared DiabeDetect with respect to several machine learning approach where we observe that DiabeDetect has performed fairly well with respect to others. An analysis for non-linear classifier can be deduced with respect to the linearity of the data which is why DiabeDetect performs better. The comparison with respect to accuracy, precision, and recall is provided in Fig. 6.

5 Conclusion

Diabetes is a disease that has taken a heavy toll on human life and our contribution in this research work encompasses the diagnosis and prognosis of the disease thus enabling the caregivers to persuade prompt life saving decisions. The research work can be extended through increasing the dataset size, fine tuning the parameters and

Fig. 6 Comparative analysis with respect to DiabeDetect

Experimental Results of Accuracy, Precision and Recall value



metrics or through complex deep neural network architectures. Approach can also be taken to explore few shot learning or federated learning to investigate the diabetes recognition task. Our work encompasses a rationale and motivation behind the study, followed by related work and curating the approach which was described in detailed documentation and for which the code is available in Github.²

References

1. M.A. Ahmad, C. Eckert, A. Teredesai, Interpretable machine learning in healthcare, in *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics* (2018), pp. 559–560
2. K. Alpan, G.S. Ilgi, Classification of diabetes dataset with data mining techniques by using weka approach, in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* (IEEE, 2020), pp. 1–7
3. J.A. ALzubi, B. Bharathikannan, S. Tanwar, R. Manikandan, A. Khanna, C. Thaventhiran, Boosted neural network ensemble classification for lung cancer disease diagnosis. *Appl. Soft Comput.* **80**, 579–591 (2019)
4. F. Anowar, M. Ashraf, A. Islam, E. Ahmed, A.I. Chowdhury, A review on diabetes self-management applications for android smartphones: Perspective of developing countries, in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)* (IEEE, 2020), pp. 1–5
5. D.K. Choubey, S. Paul, S. Kumar, S. Kumar, Classification of pima indian diabetes dataset using naive bayes with genetic algorithm as an attribute selection, in *Communication and Computing Systems: Proceedings of the International Conference on Communication and Computing System (ICCCS 2016)* (2017), pp. 451–455
6. Y.A. Christobel, P. Sivaprakasam, A new classwise k nearest neighbor (cknn) method for the classification of diabetes dataset. *Int. J. Eng. Adv. Technol.* **2**(3), 200–396 (2013)
7. E. Cure, M.C. Cure, Angiotensin-converting enzyme inhibitors and angiotensin receptor blockers may be harmful in patients with diabetes during covid-19 pandemic. *Diabetes Metab. Syndr. Clin. Res. Rev.* (2020)
8. Y. Freund, L. Mason, The alternating decision tree learning algorithm. *ICML* **99**, 124–133 (1999)

² <https://github.com/militaryarman/diabedetect>.

9. M. Ghassemi, T. Naumann, P. Schulam, A.L. Beam, R. Ranganath, Opportunities in machine learning for healthcare (2018), [arXiv:1806.00388](https://arxiv.org/abs/1806.00388)
10. A. Islam, M.M. Rahman, E. Ahmed, F. Arafat, M.F. Rabby, Adaptive feature selection and classification of colon cancer from gene expression data: an ensemble learning approach, in *Proceedings of the International Conference on Computing Advancements* (2020), pp. 1–7
11. M.F. Islam, R. Ferdousi, S. Rahman, H.Y. Bushra, Likelihood prediction of diabetes at early stage using data mining techniques, in *Computer Vision and Machine Intelligence in Medical Image Analysis* (Springer, 2020), pp. 113–125
12. O. Karan, C. Bayraktar, H. Gümuşkaya, B. Karlık, Diagnosing diabetes using neural networks on small mobile devices. *Expert Syst. Appl.* **39**(1), 54–60 (2012)
13. S. Karatsiolis, C.N. Schizas, Region based support vector machine algorithm for medical diagnosis on pima indian diabetes dataset, in *2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE)* (IEEE, 2012), pp. 139–144
14. U. Kose, O. Deperlioglu, J. Alzubi, B. Patrut, Deep learning for medical decision support systems (2020)
15. J. Kupersmith, J. Francis, E. Kerr, S. Krein, L. Pogach, R.M. Kolodner, J.B. Perlin, Advancing evidence-based care for diabetes: lessons from the veterans health administration: a highly regarded ehr system is but one contributor to the quality transformation of the vha since the mid-1990s. *Health Aff.* **26**(Suppl1), w156–w168 (2007)
16. S. Lekkas, L. Mikhailov, Evolving fuzzy medical diagnosis of pima Indians diabetes and of dermatological diseases. *Artif. Intell. Med.* **50**(2), 117–126 (2010)
17. H. Naz, S. Ahuja, Deep learning approach for diabetes prediction using pima Indian dataset. *J. Diabetes Metab. Disord.* **19**(1), 391–403 (2020)
18. K. Ogurtsova, J. da Rocha Fernandes, Y. Huang, U. Linnenkamp, L. Guariguata, N.H. Cho, D. Cavan, J. Shaw, L. Makaroff, Idf diabetes atlas: global estimates for the prevalence of diabetes for 2015 and 2040. *Diabetes Res. Clin. Pract.* **128**, 40–50 (2017)
19. W.H. Organization et al., Global report on diabetes (2016)
20. S.S. Rani, J.A. Alzubi, S. Lakshmanaprabu, D. Gupta, R. Manikandan, Optimal users based secure data transmission on the internet of healthcare things (ioht) with lightweight block ciphers. *Mult. Tools Appl.*, 1–20 (2019)
21. C. Senteio, T. Veinot, J. Adler-Milstein, C. Richardson, Physicians perceptions of the impact of the ehr on the collection and retrieval of psychosocial information in outpatient diabetes care. *Int. J. Med. Inf.* **113**, 9–16 (2018)
22. V.L. Starr, A. Convit, Diabetes, sugar-coated but harmful to the brain. *Curr. Opin. Pharmacol.* **7**(6), 638–642 (2007)
23. S. Suthaharan, Decision tree learning, in *Machine Learning Models and Algorithms for Big Data Classification* (Springer, 2016), pp. 237–269
24. H. Temurtas, N. Yumusak, F. Temurtas, A comparative study on diabetes disease diagnosis using neural networks. *Expert Syst. Appl.* **36**(4), 8610–8615 (2009)
25. (US), N.D.D.G., of Diabetes, N.I., Digestive, (US), K.D., Diabetes in America. No. 95, National Institutes of Health, National Institute of Diabetes and Digestive (1995)
26. R. Vaishali, R. Sasikala, S. Ramasubbareddy, S. Remya, S. Nalluri, Genetic algorithm based feature selection and moe fuzzy classification algorithm on pima indians diabetes dataset, in *2017 International Conference on Computing Networking and Informatics (ICCNI)* (IEEE, 2017), pp. 1–5
27. G. Williams, J.C. Pickup, Handbook of diabetes (Blackwell Publication, 2004)
28. S. Zobaed, M.A. Salehi, A. Zomaya, S. Sakr, Big Data in the cloud (2019)

Waste Segregator: An Optimized Neural Learning Approach Towards Real-Time Object Classification



Drishti Singh , E. Manoj , and T. Anjali

Abstract In the present era, natural components such as air, water and land tend to act as pollutant collectors. On the road to proper waste management, human beings have developed multiple methods, and most of these require the categorical separation of waste at the earliest possible stage. This paper on a smaller scale focuses on identifying and classifying the waste into two broad categories, recyclable and organic. We used a dataset that contains nearly 22,000 images including all the classes. The model used for the work was convolutional neural network (CNN) and can be optimized more by finding optimal hyperparameters. The CNN consists of two convolutional layers followed by max pooling, fully connected layer, and an efficient GPU to make the training faster. An accuracy of 92% and above was observed with the given dataset using CNN. The experiments conducted prove that CNN is more efficient than support vector machines (SVM); however, SVM tends to give better results on smaller datasets, depending on how the data is partitioned for training.

Keywords Waste classification · Classification · Deep learning · Recycling · Machine learning · Convolutional neural network

1 Introduction

A clean person is not the one who keeps himself clean but is the one who promises to keep their surroundings clean. In India, 77% of the waste produced is disposed of in dumps, 18% is decayed and only 5% is recycled [1]. Most of the recyclable and organic waste is dumped out due to improper waste sorting. Due to rapid digitalization and urbanization, many countries across the world are facing pollution control challenges including waste management. Millions of tons of waste are generated

D. Singh · E. Manoj · T. Anjali

Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, Kerala, India

T. Anjali
e-mail: anjalit@am.amrita.edu

every day across various cities and towns out of which only a certain amount is collected and in that also very few percent are properly treated, and rest is dumped in landfill sites. The key to waste management lies in the proper segregation of waste. The first step to the proper segregation of the waste lies in sorting waste according to the proper categories. However, due to lack of awareness and resources, most of the people fail to accomplish this step, which is throwing the waste into the proper dustbin which ultimately leads to a more complex level of segregation of waste and makes it difficult on next steps.

A structured waste management system was implemented in the Kollam district of Kerala and this model could be used in making it more effective [2].

There have been multiple technological advancements at the higher level of segregation of waste which includes processing and treating and further disposing of the waste. However, the processing step goes wrong if different waste types are intermixed together. And manual sorting can be potentially harmful. The fundamental problem which was the motivator for the project was improper sorting of disposal of waste into recyclable and organic bins. Even though all products have a label that describes the type of waste it is, it is still missorted.

To fix this problem, we came up with an innovative idea of image classification of waste by using machine learning. Deep learning allows the machine to learn without human supervision by gathering information from different sources [3]. The basic model developed can further be implemented in an android application which uses the camera to detect the image and classify based on the model or an all-time video camera could be installed over the dustbins where, when the garbage is placed in front of the camera displays the type of waste in real time. Implementing the model in real time would improve the efficiency as the model would learn more from the data it collects daily. We are pursuing a neural network approach to complete the task of image classification using a multilayer convolutional neural network (CNN). CNN has proved to be successful in training and image classification compared to the other machine learning algorithms. The input which would be fed to the model would be a jpeg image as part of the dataset and the output would be the prediction class of the image (Recyclable or Organic). This paper aims on achieving a better accuracy for the Recyclable-Organic dataset and can be further implemented in various fields on larger scales and prove to be beneficial to the entire community.

2 Related Works

Yang and Thung [4] proposed a neural network approach where they used Support Vector Machine or SVM with scale-invariant transforms which find blob-like features in an image and describe each in 128 numbers. They created their datasets and classified them into six different categories-cardboard, paper, plastic, metal, glass and trash. They achieved a test accuracy of 63% using a 70–30 training testing data split and achieved 22% classification accuracy for CNN. However, this CNN was not trained to full capacity due to difficulty in finding optimal hyperparameters and

minimal data source. There is another highly capable CNN architecture that became a basic approach for image classification [5].

Another similar project was—“Auto Trash” which classifies the waste into two categories either compost or recyclable, the project was built using TensorFlow and had hardware components as well [6]. Another waste sorting related project was an application designed for contemptuously segmenting a heap of garbage in an image. The primary goal was to be able to identify the junk and report it to the officials. The authors trained the Image Net model and obtained an accuracy of 87.69% [7].

Faster R-CNN is also used for object detection method that includes Region Proposal Networks (RPNs) which shares the convolutional layers with object detection networks [8, 9].

Recycling-based classification problems were based on classification using the features of an object [10]. Another image classification performed on a different dataset using SIFT, Reflectance-based features, Micro Texture, and Bayesian framework proved helpful in optimizing the model [11].

Rectified activation units (rectifiers) are essential for state-of-the-art neural networks which enable us to train deep rectified models directly from scratch [12]. This paper provided a good envision for our project.

A content-based image retrieval methodology used in the classification approach also provided insight for feature extraction as it is dependent on the user retrieval phase and then on pattern classification [13]. The image classification done using support vector machine proves to be a helpful insight while implementing the SVM for our model as it provided more satisfactory results compared to existing SVM systems. However, CNN provides better results when compared to SVM for our dataset [14]. Coalition-based Ensemble Design (CED), an algorithm for optimal classifier ensemble also provides a wide range of comparison among different classification algorithm which helps in finding the preferred accuracy for given dataset [15].

3 Proposed Methodology

The overall architecture of our neural network contains two layers and is a defined sequential model. Neurons with weights and bias are the building block of the convolutional network. Each neuron takes the input and passes through the activation function after the weighted sum is calculated and gives back the output for further calculations.

3.1 Convolutional Neural Network

The first layer in the convolutional neural network is the convolutional layer. In this layer, the 3D filter is converted to a 2D filter by taking a filter from each element

of the matrix and finally summing it for the sliding action. Each image would pass through the convolutional layer.

Layer 1: Convolutional Layer

Convolutional layer filters with 3×3 size. The activation function used is a rectified linear activation function or ReLU which outputs the positive output same as input for values greater than zero else gives zero as the output is performed after performing convolution.

$$R(x) = \max(0, x) \quad (1)$$

Layer 2: Max Pooling Layer

Max pooling helps to select the highlighted pixels from the image as compared to the average pooling which smooths out the image and hence some features might be ignored when the average pooling method is used (Fig. 1).

Layer 3: 2nd Convolutional Layer

2nd Convolutional Layer has 32 filters with 3×3 size. The activation function used is a rectified linear activation function to increase the non-linearity in the function (Fig. 2).

Layer 4: Max Pooling Layer

Max pooling function selects the maximum element from the matrix of feature map covered by the filter and hence the output would be the most eminent feature of the previous layer feature map. The filter used for max pooling is 2×2 . The results are down sampled feature maps highlighting the dominant feature in the map.

Layer 5: Flattening

In flattening the data is converted into a one-dimensional array which is used as the input for the next layer we have flattened the output of the pooling and convolutional layers to create a feature array which is finally connected to the classification model known as the fully connected layer.

Layer 6: Fully Connected Layer

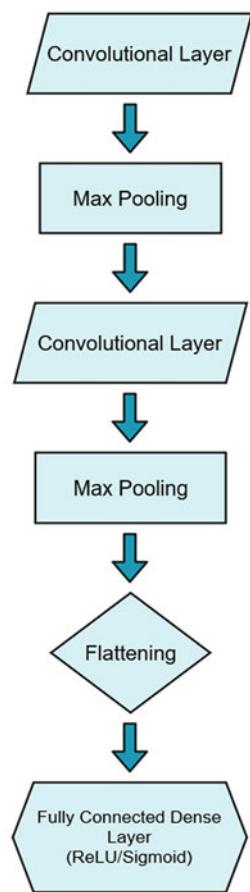
The activation function used in the dense function of a fully connected layer is a rectified linear unit function as mentioned above.

Layer 7: Fully Connected Layer

The activation function used in the dense function of a fully connected layer is a sigmoid function.

$$\frac{1}{1 + e^{-x}} \quad (2)$$

Fig. 1 Level diagram for convolutional neural network



Layers Of Convolutional Network

The best performance was achieved by using the Adam optimizer which is an extension to the stochastic gradient descent and is extremely useful in computer vision due to the process of updating network weights in the training data. The formula for moving averages of gradient and squared gradient is

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (3)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (4)$$

The hyperparameters for the model were tested and the ones with the best result were selected for the model (Table 1).

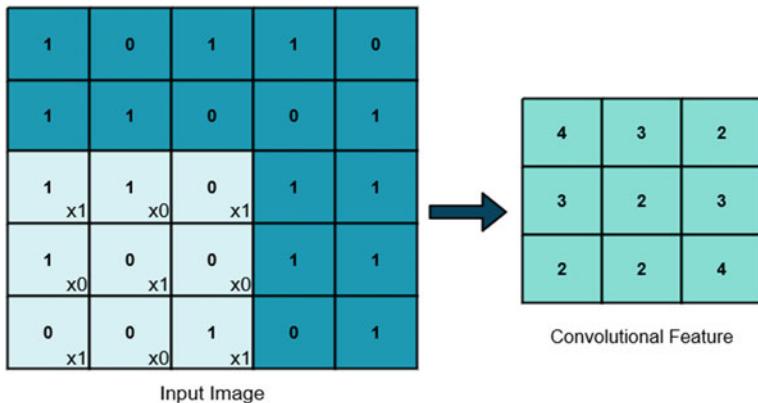


Fig. 2 Convolutional feature mapping

Table 1 CNN model summary

Layer (Type)	Output shape	Param
conv2D	(none, 62, 62, 32)	896
max pooling 2d	(none, 62, 62, 32)	0
conv2D 1	(none, 29, 29, 32)	9248
max pooling 2d 1	(none, 14, 14, 32)	0
flatten	(none, 6272)	0
dense	(none, 128)	802,944
dense 1	(none, 1)	1

3.2 Support Vector Machine

In SVM, we used a radial basis function kernel (RBF) for tuning the hyperparameters, we tried testing with the other kernels, but RBF gave the most optimized result compared to all other kernels.

Feature extraction in CNN is comparatively easier as in the case of SVM because in CNN convolutional layer manages the feature extraction along with pooling operation. For finding the hyperparameters for our SVM models we used the Grid-Search method which results in better accuracy predictions where the kernel was chosen as RBF and linear and the best value of the C parameter is from 1, 10, 100, 1000. The data was split into test and train with a split ratio of 0.3 and results were obtained after the training. A low C parameter did not tend to work well for a huge dataset (Table 2).

Table 2 SVM accuracy data

Input	Precision	Recall
Class 1 (Organic)	0.63	0.21
Class 2 (Recyclable)	0.62	0.91

Fig. 3 Recyclable object

4 Image Dataset

CNN is a cutting-edge technique for extracting features from the image and it works well only when we have sufficiently large data. CNN may tend to overfit for small datasets. Our dataset contains 22,564 train images and 2514 test images. In consideration of the dataset, we searched online and ended up with various datasets including Mindy Yang and Gary Thung's dataset which consisted of approximately 2000 images. However, for our project, we chose the dataset owned by Sashaank Sekar with a total of 22,500 images of organic and recyclable objects [16].

The dataset contains images classified as organic and recyclable. The figure below are examples of some images from the dataset from two different classes, data preprocessing was performed on each image because of different sizes and variations. For cleaning the data and making it ready to feed into the network, each image in the dataset is rescaled using the image fit generator and resized to a target shape. If the image set is fed into the network without data preprocessing the model will perform or but it will affect the performance of the model (Figs. 3 and 4).

5 Results

The work done shows how CNN, because of its complex nature, performs extremely well (approx. 92% accuracy) on a large dataset compared to SVM. On the contrary, SVM tends to give more accurate test results (approx. 85% accuracy) while using smaller datasets. For better performance, the image size was reduced along with the batch size for more appropriate results and better model training (Figs. 5 and 6).



Fig. 4 Organic object

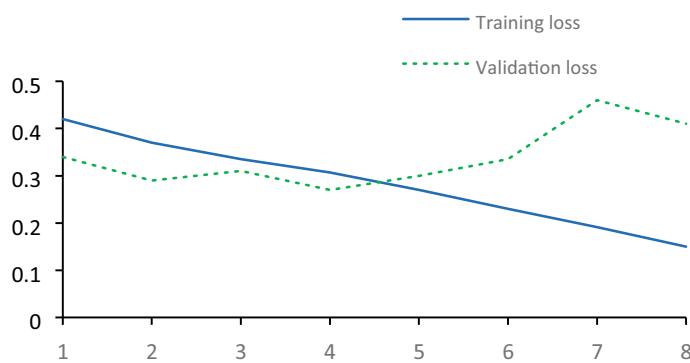


Fig. 5 Training and validation loss for CNN

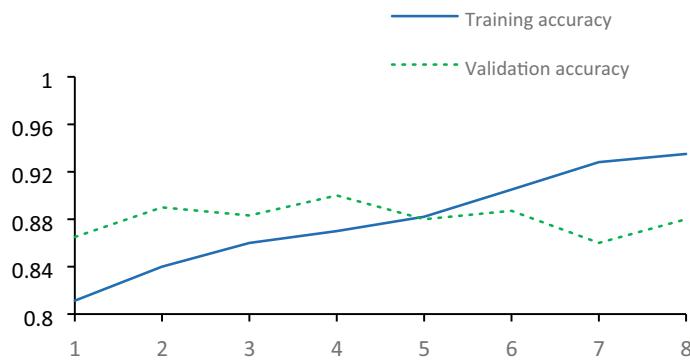


Fig. 6 Training and validation accuracy for CNN

Optimal accuracy can be achieved through CNN by increasing time for training, the number of epochs, and tuning the hyperparameters. A plausible method to improve obtained results would be to modify the hyperparameters and collect live data for training the model. However, it can be concluded that training with SVM is

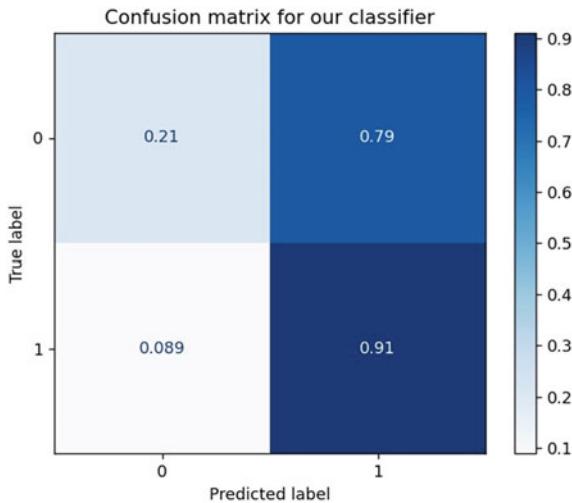


Fig. 7 Confusion matrix for SVM model



Fig. 8 Correctly predicted images by CNN model

slower than CNN in real time, with the dataset used, SVM the accuracy is inversely proportional to the size of the dataset. The finally achieved accuracy is approximately 92% using CNN after training with nearly 22,000 images (Fig. 7).

The classification of data using machine learning algorithms makes the task easy. Our results improved, as we increased the training time for the dataset. The figure below shows the correctly classified and non-classified images (Fig. 8).

6 Future Work

We would like to extend our work by improving the accuracy to better rates, by increasing the dataset and adding more real-time images to the dataset. We would like to implement the algorithm on an application for being able to detect images and

classify them in real time which can further be installed over the waste-collecting bins. We plan to use convolutional networks to extract images from video sequences for real-time classification.

Another objective would be to classify images into more categories. The algorithm could be used to sort the waste inside the bin by creating a partition that would detect the type of waste and move it to the correct partition using image classification [17].

Acknowledgements We would like to offer our sincere gratitude to our Chancellor, Dr. Mata Aritanadamayi Devi for guiding us throughout this work. It is because of her guidance and support; we were able to complete this project successfully.

References

1. <https://timesofindia.indiatimes.com/india/in-30-years-india-tipped-to-double-the-amount-of-waste-it-generates/articleshow/74454382.cms>
2. L.M. Goris, M.T. Harish, R.R. Bhavani A system design for solid waste management: a case study of an implementation in Kerala, in *2017 IEEE Region 10 Symposium (TENSYMP)* (IEEE, 2017), pp. 1–5
3. S. Tamuly, C. Jyotsna, J. Amudha, Deep learning model for image classification, in *International Conference on Computational Vision and Bio Inspired Computing* (Springer, Cham, 2019), pp. 312–320
4. M. Yang, G. Thung, Classification of trash for recyclability status. CS229 Project Report (2016)
5. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
6. J. Donovan, Auto-trash sorts garbage automatically at the TechCrunch disrupt hackathon. TechCrunch Disrupt Hackathon, San Francisco, CA, USA, Tech. Rep. Disrupt SF, 2016 (2016)
7. G. Mittal, K.B. Yagnik, M. Garg, N.C. Krishnan, Spot garbage: smartphone app to detect garbage using deep learning, in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016), pp. 940–945
8. R. Girshick, Fast R-CNN, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1440–1448
9. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1137–1149 (2016)
10. S. Zhang, E. Forssberg, Intelligent liberation and classification of electronic scrap. Powder Technol. **105**(1–3), 295–301 (1999)
11. C. Liu, L. Sharan, E.H. Adelson, R. Rosenholtz, Exploring features in a Bayesian framework for material recognition, in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (IEEE, 2010), pp. 239–246
12. K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1026–1034
13. T. Anjali, N. Rakesh, K.M. Akshay, A novel based decision tree for content-based image retrieval: an optimal classification approach, in *2018 International Conference on Communication and Signal Processing (ICCP)* (IEEE, 2018), pp. 0698–0704
14. C. Dev, K. Kumar, A. Palathil, T. Anjali, V. Panicker, Machine learning based approach for detection of lung cancer in DICOM CT image, in *Ambient Communications and Computer Systems* (Springer, Singapore, 2019), pp. 161–173
15. J.A. Alzubi, Optimal classifier ensemble design based on cooperative game theory. Res. J. Appl. Sci. Eng. Technol. **11**(12), 13361343 (2015)

16. Sashaank Sekar, <https://www.kaggle.com/techsash/wasteclassification-data/metadata>
17. A. Praveen, R. Radhika, M.U. Rammohan, D. Sidharth, S. Ambat, T. Anjali IoT based Smart Bin: a Swachh-Bharat Initiative, in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)* (IEEE, 2020), pp. 783–786

Analyzing the Impact of Forensic Accounting in the Detection of Financial Fraud: The Mediating Role of Artificial Intelligence



Kamakshi Mehta, Prabhat Mittal, Pankaj Kumar Gupta, and J. K. Tandon

Abstract In the recent times, the financial sector has witnessed number of financial scams in the markets raising concerns for the professionals and the auditors accountable for ensuring transparency and accuracy in the day-to-day auditing activities. Globally, the use of artificial intelligence has prominently increased the ability of a machine combined with human intelligence and has added to the ability of traditional methods to deal with financial information manipulation. The study aims to identify the mediating role of artificial intelligence tools in the forensic accounting and the detection of financial frauds. The study has developed a research model using structural equation modeling to examine the influence of forensic accounting in the detection of financial frauds. The results can help professionals understanding the benefits of AI technology in the forensic accounting domain that can facilitate combating fraud.

Keywords Artificial intelligence · Fraud detection · Forensic accounting · Structural equation modeling · Smart PLS

1 Introduction

The integration of techniques of investigative audit in the day-to-day accounting activities has emerged as a Novato expertise “Forensic Accounting and Audit” focusing on detection and prevention of financial frauds [12, 20]. Forensic auditing is the act involving the investigation and reporting of historical financial data for providing evidence in the court of law in cases of legal disputes [18]. Researchers

K. Mehta (✉) · J. K. Tandon
Jaipur National University, Jaipur, Rajasthan, India

P. Mittal
Satyawati College (Evening), University of Delhi, Delhi, India
e-mail: p.mittal@satyawatiedu.ac.in

P. K. Gupta
Jamia Millia Islamia University, New Delhi, India

define the term forensic in relation to financial facts and its manipulation or the presence of a financial fraud. According to Oberholzer, forensic auditing is a form of investigative audit or accounting which focuses on tracing financial fraudulent transactions.

Globally, the increasing trends in the fraudulent activities have become a national threat to the countries around the world. According to ACFE [2] has estimated a loss of nearly \$ 4 tr on the global level. Rising white collar crimes have plagued the financial markets all over the world and have drawn attention toward fraud detection through a relatively new field, i.e., forensic accounting [1, 19, 25].

Forensic accounting as a field provides a deep insight relating to the frauds that take place along with preventing frauds and taking anti-fraud measures [9, 13, 17]. The advent of information technology in the current times has raised concerns of academics and professionals to implement the benefits of AI in forensic accounting for the purpose of combating financial fraud [29].

The “artificial vs. human intelligence” dilemma widely debated among academics and practitioners, encompasses many controversial issues related to the prospects of some occupations [3, 6], the required new skill set and competences, the way humans and machines could work efficiently and effectively together [15]. Many researchers argue that awareness of AI tools among the stakeholders of the account domain like auditors, analysts will take the maximum benefits out of it [11, 29]. These studies also highlight that the use of AI can empower auditors to scrutinize and transparency of the data with full testing for availability of the client [4]. The professionals are enabled to take decisions strategically and manage the risk of the organization for the better future. With the increase in volume of data in the accounting entries and business transactions, the use of AI tools has become advent and the nature of auditors’ job has changed dramatically in the forensic accounting. However, the emergence and the use of AI in forensic accounting for the detection are very scarce in small organization. The present study analyzes the relationship of the perceived usefulness and the behavioral intentions to use AI in fraud detection.

The study intends to study the mediating role of artificial intelligence tools in implementation of forensic accounting in the course of detection of the financial frauds in corporate sector and the banking industry. A structural equation modeling has been used to examine the extent of impact of the mediating variable in comparison to the direct role of awareness of forensic accounting on reduction in corporate and financial frauds.

2 Literature Review

The author believes that techniques based on big data and artificial intelligence would transform the shape and use of data in the government sector [15]. According to the researcher the current business scenario has warranted that big data and AI have become an indispensable tool in an environment where the role of technologies

such as telecommunication, social media, cloud computing, and artificial intelligence plays a pivotal role [5, 21, 23]. The study strengthens the argument that the transformation of data into information has been made possible due to the technological advancement in the field of AI [29]. The research concluded that the gap between the benefits of AI enabled FAS and its actual implementation was majorly being contributed by the resistance of accountants due to their lack of comfort zone with latest techniques. The study claims that the AI techniques are undoubtedly very important in the field of accounting and auditing. The laws and procedures of accounting also need to be upgraded so as to broaden the scope and include the AI technologies within their preview [1]. The researcher in his research paper reviewed the available literature published in Brazilian Journal during the years 2006–2015 and he concluded that very little research had been conducted till that date [17]. The author has concluded that the Brazilian auditors and professionals have understood the importance of AI techniques over the traditional accounting methods when it comes to safeguarding the assets of their organizations [29]. The use of latest AI techniques in accounting and auditing has changed the scope and method of interpretation of financial data [8]. The study reveals that the professionals when adopting the AI techniques in their accounting and auditing activities have to be very careful. As any error due to poor implementation shall result in erroneous and distorted results [12].

3 Research Design and Methodology

The research population consists of auditing professionals and employees of public offices (departments and agencies) on the grounds that they are important members of the senior management team participating in the main decisions of the organization. The Instruments of research included a standardized questionnaire with 54 factors, this questionnaire had 10 parts. Variables that were used in this study were dependent and independent ones. A total of 350 individuals are connected and received a final response from 321 with a success rate of 91.71%. The mean age of the respondents was 1.88 (S.D.: 0.331). Out of the total respondents, 71% were male, 20.6% were female and 8.4% did not prefer to share their gender.

4 Data Analysis and Results

The responses collected using the survey instrument have been analyzed with descriptive measures presented in Table 1. For analyzing the relationship and impact of big data technologies on reduction in frauds, the present study has used a non-parametric Partial Least Squares (PLS)-Structural Equation Modeling (SEM) developed by [27]. PLS-SEM has the ability to handle both formative and reflective indicators in contrast

Table 1 Construct validity and reliability

Construct	Indicator	Outer loading composite	Reliability	AVE
	PreBen1	0.815*		
Awareness of forensic counting services	Ac-PreBen2	0.708*	0.835	0.560
	PreBen3	0.675*		
	PreBen4	0.787*		
Reduction in frauds	Red. In Fraud 1	0.693*		
	Red. In Fraud 2	0.656*	0.888	0.545
	Red. In Fraud 3	0.641*		
	Red. In Fraud 5	0.701*		
	Red. In Fraud 6	0.628*		
	Red. In Fraud 7	0.574*		
	Red. In Fraud 8	0.818*		
	Red. In Fraud 9	0.664*		
	Red. In Fraud 10	0.633*		
	Tech 1	0.63*		
AI tools	Tech 2	0.753*		
	Tech 3	0.859	0.886	0.567
	Tech 4	0.776*		
	Tech 5	0.786*		
	Tech 6	0.692*		

* Significant at $p < 0.05$

to other SEM techniques. The advantage of using PLS is that it does not assume multivariate normality and has ability to handle multicollinearity among the independents [26]. The evaluation of SEM has been carried out in two stages: a measurement model to establish construct validity and reliability and secondly, the assessment of the structural model to establish the causal relationship between the constructs.

4.1 Construct Validity and Reliability

Table 1 reports the factor loadings, composite reliability, and the average variance extraction (AVE) of all latent variables to examine the internal consistency. The results indicate that all factor loadings on their respective latent variables are statistically significant [7] and found adequate based on its high composite reliability which is greater than 0.75 as suggested by [14]. Average variance extracted (AVE) values are also adequate (>0.5) as suggested by [7] and confirm the convergent validity. It can be summarized that all the indicators of the respective latent variables are highly consistent and fulfilled the minimum factor loadings, composite reliability, and the average

Table 2 **a** Discriminant validity (Fornell and Larcker Criterion). **b** Discriminant validity (HTMT Criterion)

a			
Construct	Awareness of FA services	AI technologies	Reduction in frauds
Awareness of FA services	0.748		
AI Technologies	0.646	0.753	
Reduction in frauds	0.827	0.624	0.667

b			
Construct	Awareness of FA services	AI technologies	Reduction in frauds
Awareness of FA services			
AI Technologies	0.787		
Reduction in frauds	1.029	0.711	

variance extraction values. Table 2a presents the evidence of collinearity among the constructs using the discriminant analysis. The table of discriminant analysis displays the AVE for each construct at the diagonals of the matrix and the absolute correlations of the other constructs (the non-diagonal values). It can be observed that all the diagonal values are higher than the absolute of non-diagonal values of the matrix, showing sufficient evidence of discriminant validity, i.e., the measures of constructs are not related to each other [7, 10]. Table 2b presents the heterotrait-monotrait ratio of correlations (HTMT) to assess the discriminant validity, which for all constructs is less than the threshold value 0.9 [24].

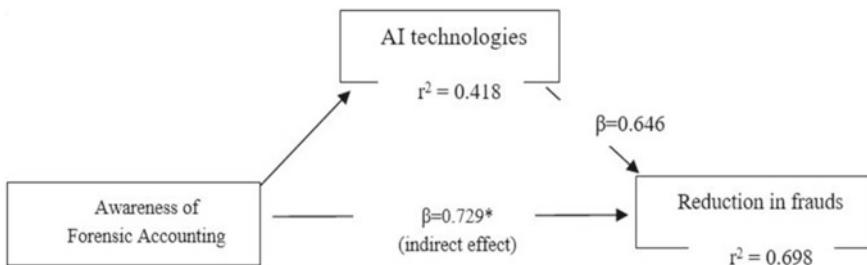
4.2 Evaluation of Structural Model

SEM model evaluates the relationship between awareness of forensic accounting services and big data technologies (exogenous variables) and reduction in frauds (endogenous variable). The study has examined three relationships, viz., the relationship between awareness level of forensic accounting (exogenous variable) and the reductions in fraud (endogenous variable), the relationship between AI technologies and reductions in fraud and finally the mediation effect of AI technology between the awareness level of FA and reduction in frauds.

The estimate of the standardized regression coefficient (β value) from AI technologies on reduction in frauds has resulted in 0.646 ($p < 0.05$) and confirms a positive influence of AI technologies in reduction of frauds. The bootstrapping resampling

Table 3 Summary of results

Hypothesis	Path	Mediation model	Results
H1	AI technologies-Reduction in frauds	0.646 ($p < 0.05$)	Significant
H2	Awareness of FA – Reduction in fraud	0.152 ($p > 0.05$)	Insignificant
H3	Awareness of FA-AI technologies-Reduction in fraud	0.729 ($p < 0.05$)	Significant

**Fig. 1** Awareness of forensic accounting-AI-reduction in frauds

technique has been carried out to test the mediating effect [22] of AI technologies in the relationship between awareness level of forensic accounting and reduction in frauds. Results indicate that the direct effect in the relationship is insignificant ($\beta = 0.152$, $p > 0.05$), while the indirect effect of the relationship between awareness level of forensic accounting and reduction in fraud in presence of AI technologies are found significant ($\beta = 0.73$, $p < 0.05$) and thus confirms the full mediation of AI technologies in the relation (see Table 3 and Fig. 1).

5 Conclusions

The result findings indicate the significance of AI technologies between awareness of forensic accounting tools and reduction in frauds. The results support the findings of [8, 19] and imply that awareness among practitioners is important to monitor day-to-day financial activities and frauds along-with the use of AI technologies. AI technologies are the most important techniques to analyze the voluminous data and can be considered as a key to enhance practices and use in forensic accounting. The mediation of AI tools is surely a big advantage in detection of spurious transactions at an incredibly early stage. The study has significant implications for the government and the accounting practitioners who can come forward for arranging training programs on understanding the use of AI tools in forensic accounting. The government can also initiate to take necessary steps to make it a part of curriculum at a different level of academic and professional courses. The use of AI technology would enhance the skills of professional auditors and the accounting practitioners.

The present study also sets the direction for future research to integrate AI technology with auditing in tuned with the accounting standards.

References

1. M.A. Abdu, H. Debajie, The impact of forensic accounting on financial performance of investment firms. *Glob. Adv. Res. J. Econ. Account. Finan.* (2019), <https://doi.org/10.5281/zenodo.3714480>
2. ACFE, Global Study on Occupational Fraud and Abuse Goverment Edition. *Report to the Nations* (2018)
3. A. Arora et al., Role of emotion in excessive use of Twitter during COVID-19 imposed lockdown in India. *J. Technol. Behav. Sci.* (2020). <https://doi.org/10.1007/s41347-020-00174-3>
4. S. Azimee, J. Akhter, Financial inclusion in India—major issues. *J. Bus. Manag. Inf. Syst.* **5**(1), 16–24 (2018), <https://doi.org/10.48001/jbmis.2018.0501003>
5. B.M. Balachandran, S. Prasad, Challenges and benefits of deploying Big Data analytics in the cloud for business intelligence. *Procedia Comput. Sci.* (2017), doi:<https://doi.org/10.1016/j.procs.2017.08.138>
6. P. Chakraborty et al., Opinion of students on online education during the COVID-19 pandemic. *Hum. Behav. Emerg. Technol.* **6**, 37–39 (2020). <https://doi.org/10.1002/hbe2.240>
7. C. Fornell, D.F. Larcker, Evaluating structural equation models with unobservable variables and measurement error. *J. Mark. Res.* (1981). <https://doi.org/10.2307/3151312>
8. A. Gepp, M.K. Linnenluecke, T. Smith, Big Data in accounting and finance: a review of influential publications and a research agenda. *J. Account. Literat.* **40**, 102–115 (2018)
9. M. Gupta, Abuse of dominant position by Google LLC—a case analysis. *J. Bus. Manag. Inf. Syst.* **7**(1), 11–21 (2020). <https://doi.org/10.48001/jbmis.2020.0701002>
10. J. Hair, et al., Multivariate data analysis: a global perspective' in *Multivariate Data Analysis: A Global Perspective* (2010)
11. K. Islam, et al., Accounting information system: traditions and future directions (by using AIS in traditional organizations). *J. Internet Bank. Commer.* (2017)/j.protcy.2013.12.060
12. M. Kanchana, V. Chadda, H. Jain, Credit card fraud detection. *Int. J. Adv. Sci. Technol.* (2020), doi:<https://doi.org/10.17148/ijarcce.2016.5109>
13. A. Kumar, Disruptive technologies and impact on industry—an exploration. *J. Bus. Manag. Inf. Syst.* **7**(1), 1–10 (2020), <https://doi.org/10.48001/jbmis.2020.0701001>
14. S. Li et al., Development and validation of a measurement instrument for studying supply chain management practices. *J. Oper. Manag.* **23**(6), 618–641 (2005). <https://doi.org/10.1016/j.jom.2005.01.002>
15. P. Mittal, Big data and analytics: a data management perspective in public administration. *Int. J. Big Data Manag.* **1**(1), 1 (2020). <https://doi.org/10.1504/ijbdm.2020.10032871>
16. B.W. Morris, Forensic and investigative accounting. *Int. J. Account.* **45**(4), 496–499 (2010). <https://doi.org/10.1016/j.intacc.2010.09.007>
17. M.S. Öztürk, H. Usul, Detection of Accounting Frauds Using the Rule-Based Expert Systems within the Scope of Forensic Accounting (2020), <https://doi.org/10.1108/S1569-37592020000102013>
18. A. Parashar, Factors affecting retirement planning behavior of working individuals: a case study in Lucknow. *J. Bus. Manag. Inf. Syst.* **5**(1), 25–34 (2018), <https://doi.org/10.48001/jbmis.2018.0501004>
19. Z. Rezaee, J. Wang, Relevance of big data to forensic accounting practice and education. *Manag. Audit. J.* **34**(3), 268–288 (2019). <https://doi.org/10.1108/MAJ-08-2017-1633>
20. I. Sadgali, N. Sael, F. Benabbou, Performance of machine learning techniques in the detection of financial frauds. *Procedia Comput. Sci.* (2019), doi:<https://doi.org/10.1016/j.procs.2019.01.007>

21. A. Saxena, V.P. Bansal, CSR and Covid-19: redefining the practices of CSR in context of Indian Industries. *J. Bus. Manag. Inf. Syst.* **7**(1), 47–53 (2020), <https://doi.org/10.48001/jbmis.2020.0701005>
22. P.E. Shrout, N. Bolger, Mediation in experimental and nonexperimental studies: New procedures and recommendations. *Psychol. Methods* (2002). <https://doi.org/10.1037/1082-989X.7.4.422>
23. T.H. Siddiqui, R.K. Yadav, A study of marketing strategies of pharmaceutical industry in India. *J. Bus. Manag. Inf. Syst.* **6**(1), 27–37 (2019), <https://doi.org/10.48001/jbmis.2019.0601003>
24. T. Teo, W.S. Luan, C.C. Sing, A cross-cultural examination of the intention to use technology between Singaporean and Malaysian pre-service teachers: an application of the Technology Acceptance Model (TAM). *Educ. Technol. Soc.* (2008), <https://doi.org/10.1111/bjet.12169>
25. M. Tutino, M. Merlo, Accounting fraud: a literature review, in *Risk Governance and Control: Financial Markets and Institutions* (2019), <https://doi.org/10.22495/rgev9i1p1>
26. J.C. Westland, Confirmatory analysis with partial least squares confirmatory analysis with partial least squares. *Decis. Sci.* (2007)
27. H. Wold, The Partial Least Squares-Fix Point Method of Estimating Interdependent Systems with Latent Variables. *Commun. Stat. Theory Methods* (1981). <https://doi.org/10.1080/03610928108828062>
28. S. Yadav, P. Chakraborty, P. Mittal, User interface of a drawing app for children: design and effectiveness, in *Advances in Intelligent Systems and Computing* (2021), pp. 53–61, 10.1007/978-981-15-5113-0_4
29. A. Zemánková, Artificial intelligence and blockchain in audit and accounting: literature review. *WSEAS Trans. Bus. Econ.* **16**, 568–581 (2019)

Knowledge Discovery in Geographical Sciences—A Systematic Survey of Various Machine Learning Algorithms for Rainfall Prediction



Sheikh Amir Fayaz, Majid Zaman, and Muheet Ahmed Butt

Abstract One of the biggest challenges faced by humanity over time is weather prediction. Rainfall prediction plays a critical role in agricultural sciences, besides it is pivotal in the prediction of droughts and floods. Over the last few decades, weather, especially rain has become much more unexpected with primarily the blame being climate change. This review paper presents an introduction to the implementation of machine learning on the geographical datasets: How it started, how much has been achieved, and what is the current scenario of machine learning in geographical sciences. The main area covering in this paper are meteorological data, Ensemble models, regression, classification including Neural Networks, Support vector machine (SVM), Decision trees, Naïve Bayes, J48, CART, and ID3. However, the main thrust area has been the implementation of various neural network models which includes BPNN, FFNN, GWLM-NARX, RNN, and TDNN in geographical data sciences.

Keywords Rainfall prediction · Data mining · Neural networks · Classification · Prediction · Geographical data, meteorological data

1 Introduction

1.1 Data Mining

Data mining is the practice of examining large preexisting databases in order to generate new information, patterns, and knowledge and empower us in making appropriate decisions and also predict the future course of actions. Data Mining is a step wise process of discovery of information consisting of a set of data mining

S. A. Fayaz · M. A. Butt
Department of Computer Science, University of Kashmir, Srinagar, India
e-mail: amirfayaz.scholar@kashmiruniversity.net

M. Zaman (✉)
Directorate of IT & SS, University of Kashmir, Srinagar, India

methods and algorithms that, in acceptable limits, discover significant patterns in data structures, which will indicate the general trends. This type of mining will uncover the patterns in data using analytical methods. These patterns play a very important role in producing decisions because they highlight the areas where processes need improvement and these discovered patterns will help to make improved decisions in less time.

1.2 Need of Data Mining

Lot of raw data is available in databases which include grid databases [1], Academic & [2–6] educational databases, health databases [7–12], and metrological databases [13–16]. Manual analysis of these databases is an error prone and time-consuming process. So, data analysis will play a major role in generating the patterns from these set of raw data to predict the actual behavior of datasets. Thus, this automatic analysis of data is needed and thus there comes the need of automatic data mining.

1.3 Machine Learning

Machine learning has stolen the spotlight. It has become the number one search trend during the year 2019 and continues to grow unimaginably. Machine learning is the study of scientific algorithms and various statistical models which are used to perform a specific task, that rely on the patterns, generating inferences and interpretations. Machine Learning Algorithms builds a Model-Mathematical [17] based on the data available (Data sets), Known as “Training Data sets”, and in order to make predictions or decisions, the model generated from the training sets will result in the rules, by which the future predictions or decisions can be made.

1.4 Geographical Sciences

The term “Geographical sciences” is the systematic study of earth and is considered as one of the original disciplines along with the idea of trying to understand the world where humans live. In geographical discipline, geographers focus on the climate, climate changes, biogeography, weather forecasting, and other environmental processes. The technologies that geographers and other scientists use to study the physical and cultural environments spatially and over time are called as geospatial technologies. A geospatial technology is a broad term that incorporates global positioning system (GPS), remote sensing, and geographic information system (GIS) [18].

2 Geographical Data Mining

In today's world, concept of data is everywhere and it is increasing progressively. Huge volumes of data have been warehoused in datacenters, databases, geographic information systems (GIS), and various other storage repositories. This data has been categorized into big data and spatial data. Scientists use big data to apprehend complex "systems" such as natural processes, meteorological conditions, and climate change, whereas spatial data also called as geospatial data is a form of big data, which is linked to the specific location on Earth. This massive amount of data has posed a great task to the traditional data exploration methods for information and knowledge detection and discovery in databases (KDD). The extraction of knowledge or data mining is the field of discovering novel and potentially useful information from large amount of data. Various data mining procedures and statistical approaches are used for extracting the challenging information from these sets of data.

Data mining plays a very significant role in the extraction of knowledge from the metrological data in the field of Geographical sciences. In present period, Weather prediction and analysis has become a challenging problem around the globe from the last century, because predicting weather is advantageous for many living activities like crop growing in the field of agriculture sector, tourism in the field of travel sector, and natural disaster prevention [19, 20].

3 Machine Learning Approaches for Rainfall Prediction

Climate change analysis analyzes the behavior of weather for a definite time period. One of the important climate change task is the rainfall forecasting, where specific features such as humidity, temperature, wind, atmospheric pressure, and precipitation are used in the prediction of the rainfall of a specific location. Predicting rainfall has been a challenging task in weather analysis for every researcher in the current environment. Accurate rainfall prediction has not been possible so far, because of its complexity measures. Precise and timely rainfall prediction can lead to take various operative security procedures in advance in the ongoing development of the projects, agricultural tasks, travel situations and alarming flood situations, etc.

Data mining and machine learning approaches can be very effective in predicting the rainfall by extracting the hidden patterns from the past set of meteorological data. In this paper, we will focus on the prediction of rainfall, i.e., we review on the applications of data mining in the prediction of rainfall. By reviewing, our study will help to provide the critical analysis of the latest data mining practices and algorithms used in predicting the rainfall. Also, this will contribute the researchers in analyzing the latest work done on rainfall prediction using various data mining algorithms. Furthermore, this study will provide the future directions and comparisons of the various machine learning algorithms implemented in rainfall prediction.

4 Parameters Used in Rainfall Prediction

Since there are lot of attributes which are taken into consideration during prediction of rainfall. Many researchers have used different parameters for the prediction of rainfall depending on the area and the region. These rainfall parameters which are used for the prediction purposes mainly include maximum temperature, minimum temperature, humidity at various intervals, vapor pressure, evaporation, season, date, year, prediction-based sensors, precipitation, latitude-longitude, sea surface pressure, cloud cover, station level pressure, sea level pressure, wind speed and the quantum of rainfall, etc.

5 State of Knowledge—An Ephemeral Analysis of Rainfall Prediction

In this paper, a brief review is provided to summarize the recent studies on the weather forecasting mostly—Rainfall prediction [21–23], using various data mining techniques including supervised [24, 25], unsupervised, and semi-supervised learning algorithms, their pros and cons. Most importantly, in this paper, we will also discuss the various challenges which are yet to be faced.

To improve the precision in predicting the rainfall, scientists have been working by using various approaches which include Artificial Neural networks, Data mining and machine learning Algorithms, Fuzzy Inference systems, SVM, and many other techniques. Since a lot of literature is available for rainfall prediction, we have chosen some of the selected studies which are mentioned below

Lizhen Lu et al., gave the comparative analysis using Artificial Neural Network (ANN), Support Vector Machine (SVM), and Adaptive neuro fuzzy inference system (ANFIS) in their study. Lizhen lu et al, proposed a case study on Yaojiang watershed, south east china [26]. In their paper, they have compared the forecast model on four terms, which include:

- Different lags as modeling inputs.
- Training data sets of heavy rainfall.
- Multistep performance forecasting.
- Performance in peak values and all values.

Experimental Results: Artificial Neural Networks (ANN) achieved better results for heavy rainfall prediction as compared to other approaches in this study.

Challenges: In [26], they have mentioned about the future studies which are still required for timely warning of short term flash flood forecasting, i.e., prediction of flash flood accuracy should be improved.

M. Selva Balan et al., predicts the rainfall using deep learning on highly nonlinear data. In their study, they proposed some of the few statistical techniques and use of

artificial neural network to predict rainfall. Also, a multilayer feed forward neural network with back-propagation technique is used to reduce the error [27]. Normalized and prepossessed data have been used, which involves elimination of those attributes that do not contribute to the prediction of rainfall.

The dataset used in [27], is of Thiruvananthapuram region from the year 1978–2017, which contains 11688 samples and 9 features. This dataset is divided into 70% training, 10% validation, and 20% testing data.

Experimental Results: Normalized dataset performs better than other and has minimal loss. Thus, Artificial Neural Networks performs well with the normalized and preprocessed data in prediction.

Challenges: In [27], the proposed model can be extended to predict the rainfall in advance and evaluates the occurrence of flood, which will prove advantageous in developing an alert system.

J. Refonaa et al. introduced a novel model for monthly rainfall prediction using linear regression analysis. Various parameters have been used in the prediction, which includes: temperature, humidity, and wind [28]. The proposed model predicts the rainfall and is based on the previous dataset available, which is of a particular geographic area. The dataset used in [28], is of Chennai region. In this study, performance of model is more accurate when compared with traditional rainfall prediction system.

Experimental Results: The proposed model works fine for classification and linear regression algorithms. The model has good accuracy. Features are extracted from the training dataset and it proved to be more accurate in predicting the weather in advance. Furthermore, other parameters like computational time and efficiency were also calculated and were proven to be better than that of other systems.

Sarita Azad et al. examined the annual and monthly data to analyze monsoon rainfall prediction in India. In [29], the periodic data has been mined using wavelet transformation and artificial neural network (ANN). It was observed that the variance of 30% and 15% were estimated in periodic and random components, respectively, of the total rainfall in case of annual data, and on the other hand, the monthly data model gives 93% variance.

Experimental Results: In [29], A welch technique has been used to estimate the power spectral density (PSD) function of spectrally homogeneous regions (SHR7) annual rainfall data from 1871 to 2005 for performing the periodicities search in SHR7 rainfall data [30]. It was observed that there were only two significant periods having 95% of confidence level. Therefore, this method of prediction on the direct rainfall data was not a good choice. Thus, the decomposition of the data into various scales using welch technique of MRA and applied artificial neural network (ANN) on each scale time series shows better results. Therefore, it can be very useful to forecast the time series up-to a large extent by using the above methodology.

B. Narayanan et al. [31], predicts the rainfall using ensemble model which includes AdaSVM and AdaNaïve. These ensemble classification methods are then compared with machine learning methods (SVM and Naïve Bayes) [31]. The experiment is

carried out on the Cuddalore district, Tamil Nadu, India, where a series of historical data of around 102 years (1901–2002) has been taken. The dataset consists of various attributes, viz: year, month, temperature, cloud cover, maximum temperature, vapor pressure, minimum temperature, precipitation, etc. The accuracy of both machine learning methods (SVM and Naïve Bayes) and ensemble method (AdaSVM and AdaNaive) have been compared, and it was observed that AdaSVM and AdaNaive methods perform better than SVM and Naïve Bayes.

Experimental Results: In this [31] study, The results showed that the accuracy of the prediction of rainfall in SVM and Naïve Bayes were 97.84% and 88.96% with the classification error of 2.16% and 11.04%, respectively, and that of AdaSVM and AdaNaive were 98.66% and 97.62% with the classification error of 1.34% and 2.38%, respectively. Thus, the ensemble methods shows good amount of increase in the accuracy and decrease in the classification error in predicting the rainfall.

G. Vamsi Krishna, proposed a study, in which prediction of rainfall was carried out using satellite images. Gaussian mixture model was used in developing a segmentation algorithm. This model was divided into 2 folds, which includes database creation and prediction. Various metrics were taken under consideration for performance analysis like Peak Signal to Noise Ratio (PSNR), IF, and Mean squared error (MSE). In this study [32], K-Means algorithm was implemented for effective segmentation of the satellite images of different regions and Gaussian mixture model was carried out to classify the weather data.

Experimental Results: [32] performance has been tested in both, i.e., in the presence and absence of clustering algorithms. The model shows improved performance in the case of clustering algorithms.

Suhaila Zainudin et al. [33] proposed a comparative study on analyzing various data mining techniques for rainfall prediction. In this [33] study, the authors analyze various multiple classifiers like Naïve Bayes, SVM, Decision tree, Neural Network (NN), and Random forest (RF) for the prediction of rainfall. Malaysian rainfall dataset from January 2010 to April 2014, which includes various parameters like temperature, rainfall, relative humidity, water level, etc., have been tested for prediction. These above classifiers/algorithms were implemented on a machine learning analytical and visualization tool “WEKA 3.7”, in which the dataset has been partitioned into various ratios of testing and training sets like (10–90)%, (20–80)%, (30–70)% ... (70–30)%, ... (90–10)%. It was observed that different classifiers perform with different accuracies based on the ratio of training set and test set, i.e., Decision Tree classifier performs best in case when the training and testing set ratio was (30–70)%, respectively, with an accuracy of 73.7%. Likewise, other classifiers perform in the same manner and produced different accuracies with different training and testing ratio. In case of Neural Networks (NN), best accuracy was observed as 74.1% on (60–40)% and it was 67.3% on (20–80)% in case of NAÏVE BAYES as best accuracy. The best result achieved by support vector machine (SVM) turned out to be 67.1% with (20–80)% model ratio. Furthermore, random forest (RF) classifier achieved best accuracy of 71.9% with training and test ratio of (30–70)%, respectively.

Experimental Results: On comparing these [33] techniques for the prediction of rainfall, it was observed that Random Forest and Decision Tree performed better because these classifiers use less training data to train the model. While neural network performs much better than Decision tree and random forest, but needs more training data to train the model.

R. Venkata Ramana et al. predicts rainfall by using wavelet technique in combination with the artificial neural network (ANN) on monthly rainfall data of 44 years of Darjeeling region in India. In this study [34], authors proposed a hybrid model called as wavelet neural network (WNN) model on monthly rainfall. It was analyzed that the efficiency is above 94% when the wavelet neural network (WNN) was taken into consideration and 64% efficiency was found for artificial neural network (ANN).

Experimental Results: In this paper [34], it was observed that wavelet neural network (WNN) performs much better than artificial neural network (ANN) models.

Adil M. Bagirov et al. predicted a monthly rainfall by the combination technique, i.e., in this study [35], a cluster and regression techniques have been combined together called as Cluster wise Linear Regression (CLR) for prediction. This Cluster wise Linear Regression technique has been implemented on the past monthly rainfall data of Victoria, Australia, having 5 parameters in the dataset, from 1889 to 2014. This data was collected from 8 different geographical weather stations. Evaluation of this technique was done on by comparing the observed results with the actual results. Also, a comparison of this technique was performed with the multiple linear regression (MLR), support vector machine (SVM), and artificial neural networks (ANN). The outcome of this study demonstrates that the Cluster wise Linear Regression technique is better with respect to the other above mentioned techniques in terms of the performance of the monthly rainfall prediction.

Razeef Mohd et al., a time series prediction model has been proposed in this study [36] using Grey Wolf Levenberg–Marquardt nonlinear auto-regression with external inputs (GWLM-NARX) model. This model serves as the adaptive prediction model, where rainfall data of the former period acts as input and the results are computed using Grey Wolf Levenberg–Marquardt (GWLM) algorithm presented by NARX model. In this research, the weather input data helps to predict the rainfall automatically, where the adaptive model for the rainfall prediction is NARX neural network, which works on the error found on the predicted value and the actual data. This experiment has been carried out in matrix laboratory (MATLAB) with the weather data set of entire India and J&K state from the year 1901–2015, available from national data sharing and accessibility policy (NDSAP).

Experimental Results: In this study [36], it was observed that NARX model found to perform better when compared to other models which predict the future rainfall based on the past records. The advantage of this model is that it converges much faster and generalizes better than that of other networks. Also, it was detected that MSE and PRD of this proposed model is approximately 0.0093 and 0.207, respectively.

Deepti Gupta et al., proposes a comparative analysis of various classification of algorithms for the prediction of rainfall. In this study [37], a number of algorithms were implemented on the past rainfall, data collected online from website [38] of

New Delhi region in India. The experiment was implemented on 2245 samples of data from June 1996 to September 2014, in which the prediction of rainfall was based on five selected attributes, viz: Mean temperature, Dew Point Temperature, Humidity, Sea level Pressure, and Wind Speed, each of numeric data type. The experiment was carried out using various algorithms, which include Classification and regression tree algorithm (CART), Naïve Bayes, K-nearest neighbor, and 5-10-1 pattern recognition neural network (PRNN). In each case, the available rainfall data set was divided into training dataset and testing dataset in the ratio of 70:30, respectively, except in PRNN, where it was divided randomly.

Experimental Results: In [37], it was observed that better result were achieved by pattern recognition neural network (PRNN) with an accuracy of 82.1% followed by KNN with accuracy measure of 80.7%, then CART with 80.3 accuracy and Naïve Bayes with 78.9% accuracy measure.

A. Geetha et al. highlights machine learning model with the data mining technique for weather forecasting. In this research [39], authors use decision tree technique for prediction of rainfall. This classification technique has been implemented on the previous dataset collected from the site [40] of Trivandrum region of India, from the year (2013–2014). This raw data collected contained around 20 attributes, but after preprocessing only 12 attributes were considered for future rainfall prediction.

Rajesh Kumar (August 2013), proposed a model which can predict events like rain, fog, and thunder by the set of different parameters like temperature, humidity, and pressure. In this [41] paper, weather data has been collected from [42], in which 64 samples were used for training the model and 72 samples were used for testing purposes. Before implementation process, data was first pre-processed and normalized and it was observed that 46 samples were classified correctly out of 72 samples with the Cohen kappa value of 0.0584 and the experiment was performed in an open source tool for classification and visualization called as WEKA. Also in this paper, authors mentioned about the improvement in the accuracy by boosting or using hybrid models for weather forecasting.

Razeef Mohd et al. [13], a comparative analysis of various data mining supervised classification algorithms have been made in this study for rainfall prediction by using the historical rainfall dataset of Srinagar district of J&K, India from the year November 2015 to November 2016 (540 records), with 9 attributes collected from [42]. After the preprocessing and normalization process on the raw data, it was found that only five attributes were relevant for rainfall prediction. In this study, a comparison on numerous data mining classification algorithms were explored, which includes decision tree bases J48, random forest, Naïve Bayes, Bayes Net, Logistic Regression, IBK, PART, and bagging. The experiment has been carried out in open source classification tool WEKA with 10 folds cross validation, in which the samples were divided randomly into testing and training sets.

Results: In [13], after experiment results were compared and it was observed that Random forest has the highest accuracy of 87.76% than other classification algortihms.

Jyothis Joseph et al. (2013) presented various data mining techniques for the prediction of rainfall. The data collected in this study [43] was from Kerala region of India and it contains almost 9 years of data from January 2001 to December 2010. From this available data, monsoon rainfall data from June to September of every year from 2001 to 2010 have been analyzed. An empirical clustering and classification method was implemented in prediction of rainfall, which includes Subtractive clustering and Feed Forward Neural Network (FFNN) classification. It was found that three clusters were obtained and the accuracy was found to be around 87% with the precision and recall values of 98% and 75%, respectively.

D. F. Cook et al. (1991) in his research [44] presented a neural network in order to predict the average air temperatures. Back Propagation Neural Network (BPNN) algorithm was used for the prediction and the results were quite satisfactory.

Tao Chen et al. (1993) proposed a feature-based neural network model in order to predict rainfall. This experiment was carried in Shikoku area in Japan. A relationship between geo stationery metrological satellite (GMS) data and rainfall data with high intensity distribution was performed using four-layer neural network model. Infrared and visibility imagery of GMS image were used as input data and for learning process Back Propagation algorithm was implemented [45].

A general review of the above survey is concluded in the Table 1, in which the different approaches for rainfall forecasting are mentioned. This categorization is based on the certain parameters, which includes the publishing date (order maintained in Table 1), region, dataset, Algorithms/Model used, and the attributes present in the rainfall prediction.

Furthermore, we have also averaged the prediction accuracies for each of these techniques in order to find out the overall best and worst performing technique (Table 2 and Fig. 1). According to the results, the techniques that came up with the highest average accuracies were ANN (91.68%), KNN (92.67%), and FTDNN (89.25%), while those that came up with lowest average accuracies were Naïve Bayes (73.16%), C4.5 (75.3%), and DT (79.23).

6 Conclusion and Future Work

It is of utmost essence that right technique is employed on specific dataset, otherwise it will not produce the desired results. This research provides a complete organized mapping, as well as the comprehensive review of latest research, from 2000 till 2020, in the area of rainfall forecasting by aiming on various data mining techniques. The survey shows that BPNN, FFNN, GWLM-NARX, RNN, and TDNN are appropriate to predict rainfall than other forecasting methods such as statistical and numerical methods. In this review, a systematic research process was followed to extract and shortlist the most relevant research articles from popular digital search libraries. The research focus on the realm of rainfall prediction has been growing since last decade and so are the problem areas. An attempt was made to intellectualize the study of Knowledge discovery in meteorology using various data mining methodologies.

Table 1 General review and categorical approaches of rainfall forecasting—chronology

S. no.	Publishing year	Author(s)	Region	Dataset	Model/algorithm	Accuracy	Attributes used	References
1.	2011	G. Geetha et al.	Chennai, India	Monthly (Training set-1978 to 2009 Prediction set-1978 to 2009)	Multilayer BPNN	Satisfactory	Wind speed, mean temp., relative humidity, aerosol values	[46]
2.	2012	Soo-Yeon Ji et al.	Bowie State	(26280 Instances)	DT/4.5/CART	93–99%	—	[47]
3.	2013	Neelam Mishra	India	1871–2012	ANN, FFNN, MSE	One month regression model performs better	—	[48]
4.	2014	Haviluddin et al.	Tenggarong Station	Rainfall Data (1986–2008) 22 years	ANN/BPNN/Sigmoid Function	Satisfactory	—	[31]
5.	2015	Sarita Azad et al.	India	Rainfall Data (1877–2005) 135 years	ANN/PSD/SHR7/MRA decomposition	93–95% confidence level	—	[47]

(continued)

Table 1 (continued)

S. no.	Publishing year	Author(s)	Region	Dataset	Model/algorithm	Accuracy	Attributes used	References
6.	2016	S. Zhang et al.	Yaojiang watershed, South East China	General data	SVM/ANFIS/ANN	Neural Network performs better in High Rainfall	–	[26]
7.	2016	B. Narayanan et al.	Cuddalore district, Tamil Nadu, India	(1901–2002) 101 years	Ensemble Model (AdaSVM and AdaNaive)	AdaSVM = 98.66% AdaNaive = 97.62%	Year, month, temperature, cloud cover, max temperature, vapor pressure, min temperature, precipitation	[31]
8.	2017	Bagirov et al.	Monthly rainfall data of Victoria, Australia,	1889–2014 125 years	ANN/SVM/MLR/CLR	–	5 different rainfall parameters were used	[35]
9.	2018	Zeyi Chao et al.	Wuhan area, China	General	MEMS sensors/LSTM/BPNN/RF/ARMA/SVM	Satisfactory results	Prediction based sensors	[49]

(continued)

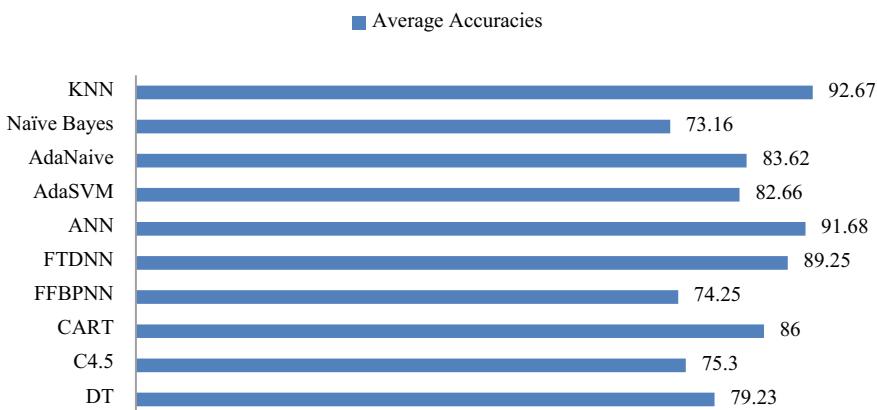
Table 1 (continued)

S. no.	Publishing year	Author(s)	Region	Dataset	Model/algorithm	Accuracy	Attributes used	References
10.	2019	M. Selva Balan et al.	Thiruvananthapuram region, India	(1978–2017) 11688 samples, dataset is divided into 70% training, 10% validation and 20% testing data	ANN, multilayer feed forward network (FFN) with back-propagation technique	Normalized dataset performs better with minimum loss	9 features/attributes	[27]
11.	2019	J. Refonaa et al.	Chennai region, India	Monthly rainfall prediction	Classification including various Neural Networks using linear regression algorithms	The model has good accuracy	Temperature, humidity, and wind	[28]
12.	2020	Razeef Mohd et al.	India and J&K state	(1901–2015) years, Available from national data sharing and accessibility policy (NDSSAP)	Grey Wolf Levenberg–Marquardt nonlinear auto-regression with external inputs (GWLM-NARX) model	NARX model	–	[36]

Table 2 Average accuracies

DT	C4.5	CART	FFBPNN	FTDNN	ANN	AdaSVM	AdaNaïve	Naïve Bayes	KNN
79.23	75.3	86	74.25	89.25	91.68	82.66	83.62	73.16	92.67

Average Accuracies

**Fig. 1** Visual representation of average accuracies

and geographic data was classified according to various data mining approaches associated with each area. It discovered how valuable, beneficial, and explosive data mining can be in geographical domain, particularly to improve prediction of rainfall, floods, etc. This study also reviewed some of the current inclinations of prediction in data mining as forecasting is concerned to be one of the contemporary themes in geographical data mining. However, it conversely remains a challenge to prognosis and warns for severe weather in a timely, accurate, and precise manner.

References

1. S.A. Fayaz, I. Altaf, A.N. Khan et al., A possible solution to grid security issue using authentication: an overview. *J. Web Eng. Technol.* **5**(3), 10–14 (2018)
2. M. Ashraf, M. Zaman, M. Ahmed, Performance analysis and different subject combinations: an empirical and analytical discourse of educational data mining, in *2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (IEEE, 2018)
3. M. Ashraf, M. Zaman, M. Ahmed, Using ensemble StackingC method and base classifiers to ameliorate prediction accuracy of pedagogical data. *Procedia Comput. Sci.* **132**, 1021–1040 (2018)
4. M. Ashraf, M. Zaman, Tools and techniques in knowledge discovery in academia: a theoretical discourse. *Int. J. Data Min. Emerg. Technol.* **7**(1), 1–9 (2017)

5. M. Ashraf, M. Zaman, M. Ahmed, Using Predictive Modeling System and Ensemble Method to Ameliorate Classification Accuracy in EDM
6. S.J. Sidiq, M. Zaman, M. Ashraf, M. Ahmed, An empirical comparison of supervised classifiers for diabetic diagnosis. *Int. J. Adv. Res. Comput. Sci.* **8**(1) (2017)
7. S. Mirza, S. Mittal, M. Zaman, Applying decision tree for prognosis of diabetes mellitus. *Int. J. Appl. Res. Inf. Technol. Comput.* **9**(1), 15–20 (2018)
8. M. Ashraf, S.M. Ahmad, N.A. Ganai, R.A. Shah, M. Zaman, S.A. Khan, A.A. Shah, Prediction of cardiovascular disease through cutting-edge deep learning technologies: an empirical study based on TENSORFLOW, PYTORCH and KERAS, in *International Conference on Innovative Computing and Communications* (Springer, Singapore, 2020), pp. 239–255
9. S. Mirza, S. Mittal, M. Zaman, Decision support predictive model for prognosis of diabetes using SMOTE and decision tree. *Int. J. Appl. Eng. Res.* **13**(11), 9277–9282 (2018)
10. M. Shuja, S. Mittal, M. Zaman, Decision support system for prognosis of diabetes using non-clinical parameters and data mining techniques. *Int. J. Database Theory Appl.* **11**(3), 39–48 (2018)
11. J.A. Alzubi, A. Kumar, O. A Alzubi, R. Manikandan, Efficient approaches for prediction of brain tumor using machine learning techniques. *Indian J. Public Health Res. Dev.* (2019). <https://doi.org/10.5958/0976-5506.2019.00298.5>
12. A. Khamparia, A. Singh, D. Anand, D. Gupta, A. Khanna, N. Arun Kumar, J. Tan, A novel deep learning based multi-model ensemble methods for prediction of neuromuscular disorders. *Neural Comput. Appl.* (Springer) (2018). <https://doi.org/10.1007/s00521-018-3896-0>
13. R. Mohd, M.A. Butt, M. Zaman, Comparative study of rainfall prediction modeling: a case study on Srinagar, J&K, India. *Asian J. Comput. Sci. Technol. (AJCST)* **6**(1) (2018). ISSN: 2249-0701
14. M. Zaman, M.A. Butt, Information translation: a practitioners approach, in *World Congress on Engineering and Computer Science (WCECS)*, San Francisco, USA, Oct 2012
15. R.M. Shah, M.A. Butt, M.Z. Baba, Review of predictive analytic modeling techniques. *Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS)* **6**(4), 58–62 (2017)
16. R. Mohammad, M. Butt Ahmed, M. Baba Zaman, Tools for predictive analytics: an overview. *Int. J. Sci. Res. Eng. Technol. (IJSRET)* **6**(7), 748–750 (2017)
17. S.A. Fayaz, I. Altaf, A.N. Khan, Z.H. Wani, A possible solution to grid security issue using authentication: an overview. *J. Web Eng. Technol.* **5**(3), 10–14 (2019)
18. <https://www.opengeography.org/ch-1-intro-to-geographic-science.html>
19. M. Ramzan Talib, T. Ullah, M. Umer Sarwar, M. Kashif Hanif, N. Ayub, Application of data mining techniques in weather data analysis. *IJCSNS Int. J. Comput. Sci. Netw. Secur.* **17**(6) (2017)
20. R. Mohd, M. Ahmed, M. Zaman, SALM-NARX: Self Adaptive LM-based NARX model for the prediction of rainfall, in *International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2018)* organized by SCAD Institute of Technology, Palladam, Tamil Nadu, India and technically sponsored by IEEE
21. M. Zaman, S. Kaul, M. Ahmed, Analytical comparison between the information gain and gini index using historical geographical data. (*IJACSA*) *Int. J. Adv. Comput. Sci. Appl.* **11**(5), 429–440 (2020)
22. S.A. Fayaz, M. Zaman, M.A. Butt, To ameliorate classification accuracy using ensemble distributed decision tree (DDT) vote approach: An empirical discourse of geographical data mining. *Procedia Comput. Sci.* **184**, 935–940 (2021)
23. How machine learning is redefining geographical science: a review of literature. *Int. J. Emerg. Technol. Innov. Res.* **6**(1), 1731–1746 (2019). <https://www.jetir.org>, ISSN:2349-5162. Available: <http://www.jetir.org/papers/JETIRDW06285.pdf>
24. M. Ashraf et al., Knowledge discovery in academia: a survey on related literature. *Int. J. Adv. Res. Comput. Sci.* **8**(1) (2017)
25. M. Ashraf, M. Zaman, M. Ahmed, To ameliorate classification accuracy using ensemble vote approach and base classifiers, in *Emerging Technologies in Data Mining and Information Security* (Springer, Singapore, 2019), pp. 321–334

26. S. Zhang, L. Lu, J. Yu, H. Zhou, Short-term water level prediction using different artificial intelligent models, in *2016 5th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2016* (2016)
27. M. Selva Balan, J.P. Selvan, H.R. Bisht, Y.A. Gadgil, I.R. Khaladkar, V.M. Lomte, Rainfall prediction using deep learning on highly non-linear data. *Int. J. Res. Eng. Sci. Manag.* **2**(3) (2019). ISSN (Online): 2581-5792
28. J. Refonaa, M. Lakshmi, R. Abbas, M. Raziullha, Rainfall prediction using regression model. *Int. J. Recent Technol. Eng. (IJRTE)* **8**(2S3) (2019). ISSN: 2277-3878
29. S. Azad, S. Debnath, M. Rajeevan, *Analyzing Predictability in Indian Monsoon Rainfall: A Data Analytic Approach* (Springer International Publishing Switzerland, 2015). *Environ. Process.* **2**, 717–727 (2015). <https://doi.org/10.1007/s40710-015-0108-0>
30. S. Azad, T. Vignesh, R. Narasimha, Periodicities in Indian monsoon rainfall over spectrally homogeneous regions. *Int. J. Climatol.* **30**, 2289–2298 (2010)
31. B. Narayanan, M. Govindarajan, Rainfall prediction based on ensemble model. *Int. J. Innov. Res. Sci. Eng. Technol. (An ISO 3297: 2007 Certified Organization)* **5**(5) (2016). ISSN (Online): 2319-8753. ISSN (Print): 2347-6710
32. G. Vamsi Krishna, Prediction of rainfall using unsupervised model based approach using K-Means algorithm. *Int. J. Math. Sci. Comput. (IJMSC)* **1**(1), 11–20 (2015). <https://doi.org/10.5815/ijmsc.2015.01.02>
33. S. Zainudin, D.S. Jasim, A.A. Bakar, Comparative analysis of data mining techniques for Malaysian rainfall prediction. *Int. J. Adv. Sci. Eng. Inf. Technol.* **6**(6) (2016). ISSN: 2088-5334
34. R. Venkata Ramana, B. Krishna, S.R. Kumar, N.G. Pandey, Monthly rainfall prediction using wavelet neural network analysis. *Water Resour. Manag.* **27**, 3697–3711 (2013). <https://doi.org/10.1007/s11269-013-0374-4>. Received: 19 Dec 2012. Accepted: 20 May 2013. Published online: 19 June 2013
35. A.M. Bagirov, A. Mahmood, A. Barton, Prediction of monthly rainfall in Victoria, Australia: clusterwise linear regression approach. *Atmos. Res.* (2017). <https://doi.org/10.1016/j.atmosres.2017.01.003>
36. R. Mohd, M.A. Butt, M.Z. Baba, GWLM–NARX. *Data Technol. Appl.* **54**(1), 85–102 (2020)
37. D. Gupta, U. Ghose, A comparative study of classification algorithms for forecasting rainfall, in *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions)*. ISBN: 978-1-4673-7231-2/15/\$31.00 ©2015 IEEE, Sept 2015. <https://doi.org/10.1109/icrito.2015.7359273> 36
38. <http://www.wunderground.com>
39. A. Geetha, G.M. Nasira, Data mining for meteorological applications: decision trees for modeling rainfall prediction, in *2014 IEEE International Conference on Computational Intelligence and Computing Research*, ISBN: 978-1-4799-3975-6/14/\$31.00 ©2014 IEEE
40. <ftp://ftp.ncdc.noaa.gov/pub/data/gsod/2014/>
41. R. Kumar, Decision tree for the weather forecasting. *Int. J. Comput. Appl.* **76**(2) (2013). ISSN: 0975-8887
42. <http://www.wundergrounds.com>
43. J. Joseph, T.K. Ratheesh, Rainfall prediction using data mining techniques. *Int. J. Comput. Appl.* **83**(8) (2013). ISSN: 0975-8887
44. D.F. Cook, M.L. Wolfe, A back-propagation neural network to predict average air temperatures. *AI Appl.* **5**, 40–46 (1991)
45. T. Chen, M. Takagi, Rainfall prediction of geostationary meteorological satellite images using artificial neural network. *IGARSS* **3**, 1247–1249 (1993)
46. G. Geetha, R.S. Selvaraj, Prediction of monthly rainfall in Chennai using back propagation neural network model. *Int. J. Eng. Sci. Technol.* **3**(1), 211–213 (2011)
47. S.-Y. Ji, S. Sharma, B. Yu, D.H. Jeong, Designing a rule-based hourly rainfall prediction model, in *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on Data Analysis*, Aug 2012

48. N. Mishra, H. Kumar Soni, S. Sharma, A.K. Upadhyay, Development and analysis of artificial neural network models for rainfall prediction by using time-series data. *Int. J. Intell. Syst. Appl.* **1**, 16–23 (2018). Published Online Jan 2018 in MECS, <http://www.mecs-press.org/>. <https://doi.org/10.5815/ijisa.2018.01.03>
49. Z. Chao, F. Pu, Y. Yin, B. Han, X. Chen, Research on real-time local rainfall prediction based on MEMS sensors. *Hindawi J. Sens.* **2018**, Article ID 6184713, 9 (2018). <https://doi.org/10.1155/2018/6184713>

Hateful Memes, Offensive or Non-offensive!



Sujata Khedkar, Priya Karsi, Devansh Ahuja, and Anshul Bahrani

Abstract Memes are a form of friendly exchange between one another and can guide one smoothly over a conflict or come to a meaningful conclusion. Detection of offensive and hate speech-related content in social media memes has been investigated in a single modality only, i.e. considering only text or only image. However, memes are content plus images. Consideration of both modalities to identify offensive memes is vital. Thus, we made a model considering this multimodal nature. The training dataset available from the Facebook challenge had pre-extracted text from a meme. However, testing an original meme, the text had to be extracted with the help of Optical Character Recognition (OCR) technology. So if the text on the meme will be not distorted rather straight, the OCR technology will give an accurate extraction of text. The text had separately been passed through a text module where different techniques like Convolutional Neural Network (CNN), Bidirectional Long Short-Term Memory (BiLSTM), and stacked LSTM have been used for detection. The image has also been passed to an image module where a CNN model, i.e. VGG16 has been used for detection. This is followed by the fusion of results from both modules via the basic concatenation technique. All of this procedure had been performed using 2 different word embeddings, i.e. GloVe and Bidirectional Encoder Representations from Transformers (BERT). The best accuracy obtained was 0.485 using BiLSTM + VGG16. Comparing the results between BERT and GloVe, GloVe beat BERT with a very minor margin, though with an even larger dataset BERT could overtake GloVe.

S. Khedkar · P. Karsi · D. Ahuja (✉) · A. Bahrani

Department of Computer Engineering, Vivekanand Education Society's Institute of Technology, Mumbai 400074, India

e-mail: 2017.devansh.ahuja@ves.ac.in

S. Khedkar

e-mail: sujata.khedkar@ves.ac.in

P. Karsi

e-mail: 2017.priya.karsi@ves.ac.in

A. Bahrani

e-mail: 2017.anshul.bahrani@ves.ac.in

Keywords Social media · Hate speech · Offensive · Non-offensive · Meme classification · Multimodal · Textual data · Image data · Fusion techniques

1 Introduction

Image and text combined give a meme. A meme is used to convey ideas from person to person. Most of the social media memes are humorous, and non-hateful created for entertainment, but an honor for one person may hurt the sentiments of another person or community. To avoid hate speech and offensive content, social media sites are taking numerous steps to identify abuse, hate speech, cyberbullying, and trolling in the post.

The most famous type of content is compared to images as pictures containing text in them. The multimodal nature of memes makes it difficult to understand them via a single modality. The examples have been shown in Fig. 1.

In Fig. 1, in every one of these model images, the content text and the picture are harmless when considered by themselves. In a meme, when a text and image are viewed together, only then the semantic content of a meme becomes mean. In this manner, it is critical to think about the two modalities to comprehend the significance or aim of the meme.

To discover the extent and importance of both modalities, we need to take a closer look at the extensive research already performed in this field, and that is what exactly our next section describes. Considering the dataset format provided by the Facebook challenge, and the research, we came up with different techniques to detect memes as offensive or non-offensive. Results obtained from different methods used were compared to gain a clear understanding. Fundamentally, our method focuses on working with images and text separately using different modules and then performing a fusion of the same to get accurate results regarding the classification of memes.



Fig. 1 Examples of memes explaining multimodal nature, reprinted from “The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes” [1] by Kiela et al. (2020)

2 Related Work

2.1 *Offensive Content in Text*

In [2], Prashanth Vijayaraghava et al. formed a dataset by performing an exploratory search on tweets. The non-hateful tweets are those phrases containing positive adjectives, a combination of swear words while hateful tweets are those phrases containing a combination of swear words with races, religions, sexual orientations, etc. They extract social, cultural, and semantic features for different modalities of data and develop their model with a late-fusion of all different modalities. In [3], Aditya Bohr et al. have shown Random Forest (RF) and Support Vector Machine (SVM) classifier on Text sentences.

2.2 *Offensive Content in Image*

In [4], Tee Connie et al. introduced six convolution layers along with two fully connected layers. The filters in each convolutional layer are expanding from 16 to 128. A max-pooling of 2×2 is placed after the 1st, 2nd, 4th, and 6th convolutional layer. ILSVRC-2013's dataset was used.

2.3 *Offensive Content in Memes*

In [5], Raul Gomez et al. have shown three different CNN + RNN models: Feature Concatenation Model (FCM), Spatial Concatenation Model (SCM), and Textual Kernels Model (TCM). The dataset used is MMHS150k.

In [6], Fan Yang et al. showed different fusion modes: Basic fusion, Gated summation, and Bilinear transformation Attention Mode sparse max and softmax. The dataset was 7 months' posts about social media, which users reported as hate speech.

In [7], Homa HosseiniMardi et al. have shown a Linear SVM classifier that gives the accuracy of 87% for cyberbullying. The dataset used contains Instagram posts and their comments labeled by humans.

In [8], Benet Oriol Sabat et al. used OCR to detect text in memes and encoded in a BERT representation and the meme image was encoded using pre-trained VGG16 CNN. The dataset was formed by hate memes from Google Images and non-hate memes from Reddit Memes Dataset.

In [9], Priya Rani et al. experimented with SVM, K-Nearest Neighbors, Multinomial Naive Bayes, and Decision Tree machine learning classifiers.

In [10], Sean MacAvaney et al. showed a Multi-view SVM model that classifies text based on the words rated in the dictionary as offensive or non-offensive. In [11], Karen Simonyan et al. designed a 16-parameterized-layered CNN with max-pooling

layers after every 3 parameterized layers were followed by 3 dense layers. The dataset used was imagined.

3 Problem Statement

Hate speech can be defined as an immediate or roundabout assault on individuals dependent on attributes, including nationality, race, ethnicity, migration status, religion, rank, sex, sex personality, sexual direction, and incapacity or infection.

We aim at developing a model to deal with the multimodal nature of the memes dataset on the Internet. The basic equation of memes is an image and some meaningful text connected with the image. Memes are harmless and humorous most of the time. Nevertheless, when images are combined with text in an ill-suited format, it may result in a hateful meme thus spreading hate in the social world.

The mission of successfully identifying whether a meme is hateful or not has become tremendously tough and equally necessary in this extensive world of the Internet. The dependence of a model solely on text or images to predict whether a meme is offensive or non-offensive is incomplete due to the examples seen in Fig. 1. Thus, to get the correct answer of the meme being offensive or non-offensive, the model should consider both modalities—text and image, i.e. it should be of multimodal nature.

4 Dataset

For this project, we have used a Hateful Memes dataset by Facebook. The hateful memes dataset is split into three files: train, test, and dev jsonl files and an image directory contain all the memes. Each line in a .json file is valid JSON consists of (Fig. 2)

Fig. 2 Example of meme for the dataset



id: unique meme identification number,
 img: the path to an image file
 label: 1-> “hateful”, 0-> “non-hateful”
 text: string inserted in the meme picture
 for example
 {“id”:79861,“img”:“imgV79861.png”,“label”:1,“text”:“white people when they learn fish swim in schools”}

5 Methodology

5.1 Models

5.1.1 Models for Textual Data

A stacked LSTM, a Bidirectional LSTM, and a CNN network have been compared for meme classification based on textual data.

LSTM—Generally, an LSTM unit comprises a basic cell whose job is to remember the values for random time intervals and three control gates, namely input gate, output gate, and a forget gate whose task is to control the inflow and outflow of data in a cell. The perk of working with an LSTM cell against a classic RNN is a cell memory unit. Combining the idea of forgetting what is stored earlier in the memory and inserting the content of the newly available information is the task of the cell vector.

In short, LSTM cannot be treated as a classic bag-of-words approach which addresses every word in context as a discrete unit and is not capable of conserving the context of every word. However, LSTM is a network that treats text data as a time sequence by applying the concept of a memory unit thus being able to preserve its context.

BiLSTM—Bidirectional Recurrent Neural Network (BRNN) connects two hidden layers of different directions to a similar yield.

Bidirectional nature helps to pass inputs in two directions, i.e. one from the future to the past and one from the past to the future. The two hidden states help at any point in time to store the information from both past as well as future. Only one BiLSTM has been used in this architecture. The classification layer has a sigmoid activation function to which the output of this layer has been connected. Thus, the output layer gives the probability of the meme being classified as offensive.

Stacked LSTM—In this approach, the model includes multiple layers of LSTM. The stacked LSTM, also known as deep LSTM, was first formulated by [12] and was applied to speech recognition problems.

In the architecture for our model, stacked LSTMs are used as feature extractors before the data is being sent to the classification layer. Word embeddings are created using a pre-trained GloVe dataset and also by BERT. The use of pre-trained word embedding leverages the contextual meaning of the word globally.

The primary explanation behind the stacking of LSTM is to take into consideration the complexity of a model. In the case of stacked LSTM, similar to a feedforward net, layers are stacked to build a hierarchical feature representation to use for different machine learning tasks. Since the information is a result from an LSTM layer, the current LSTM can build a more complex feature representation of the current input.

Presently the distinction between having another LSTM layer between the feature input and the LSTM layer and having a feedforward layer is that the latter doesn't receive any feedback from its past time step and accordingly cannot record for particular patterns that may improve the results. However, the former instead gives more intricate information examples at each layer.

5.1.2 Models for Image Data

In [11], the authors have developed a convolutional neural network, a CNN model named VGG16. This model gained 92.7% test accuracy in a dataset of 14 m images of 1000 classes, i.e. ImageNet dataset. It took weeks to train VGG16 using NVIDIA Titan Black GPU. According to [13], VGG16 beats the previous models of various competitions like ILSVRC-2012 and ILSVRC-2013 competitions.

The architecture of VGG16 has been shown in Fig. 3. Fixed size of 224×224 RGB is passed as an input to the first convolutional layer, after which it is passed into a series of convolutional layers, where there was a complete utilization of the channels that had a little open field to be able to capture the notion of all directions, i.e. of size 3×3 .

In our model, images were loaded into an array and changed into a fixed shape as per VGG16 specifications. The VGG architecture has two convolutional layers

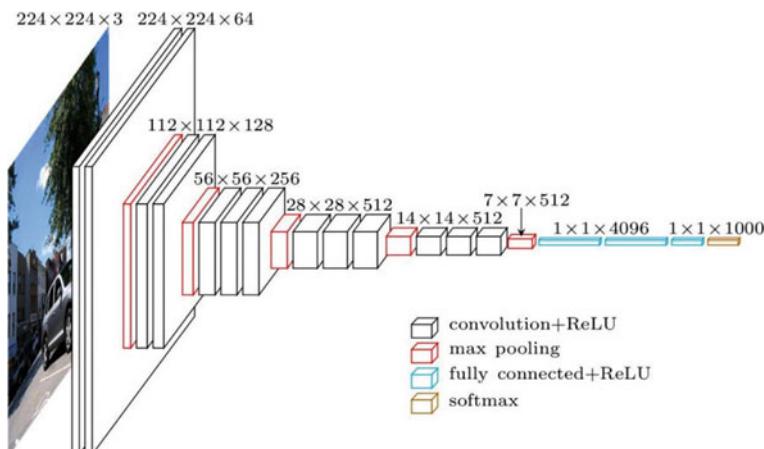


Fig. 3 VGG16 model architecture, reprinted from “VGG16—Convolutional Network for Classification and Detection” [13] by Muneeb ul Hassan, 20 November 2018

both with ReLU as an activation function. The output of the activation function has been fed to the max-pooling layer which later has been followed by a fully connected layer which also uses “ReLU” as an activation function. However, the fully connected layers have been fine-tuned toward the hatefulness of a meme. All the 16 layers in VGG16 have been frozen by converting all the parameters in the layers as untrainable. This has been done to prevent the pre-trained network again on new data. The top layer in the model, i.e. 1000 classes of ImageNet are not required and hence removed.

5.2 Overview

The whole system is divided into 2 parts, one for image classification and the other for text classification. Again, these two parts are further divided into further subsections.

The text classification section takes the text and passes it through a data cleaning module. In this module, the text undergoes data preprocessing and cleaning wherein stopwords and unimportant words are removed. On performing tokenization, padding, and text in sequence, the vector of the sentence is passed to the embedding block wherein the estimation of a word in the vector space is discovered from text and relies upon the words that incorporate the word when it is utilized. GloVe embedding and BERT have been used for the same. This vector has then been passed to the LSTM block which represents three different algorithms that have been used, BiLSTM layer, CNN layer, and Stacked LSTM layer separately, and the output has been obtained in the form whether the text is hateful or not. The results of all the three outputs received from these three layers have then been compared to gain a better understanding.

The Images Classification section takes an image and passes it through the CNN (VGG16) model initial layers. Features have been extracted based on the image only. Again, the output layer states whether the image is hateful or not.

Finally, the fusion block involves concatenation of the outputs received from the text model and image model which is then passed through a classifier block giving the output as to whether the meme is hateful or not.

The following different modules can be seen diagrammatically in Fig. 4.

5.2.1 Text Module

Data Preprocessing and Data Cleaning: This step takes the string, i.e. text of every row in the dataset, and performs the removal of stopwords and unnecessary words and symbols from the text available. It then converts the token of words into vectors of sentences.

Embedding Layer: GloVe embeddings and BERT embeddings have been used to obtain the vector representation of input words. On performing tokenization, padding, and text to sequence, the vector of the sentence is passed to the embedding block

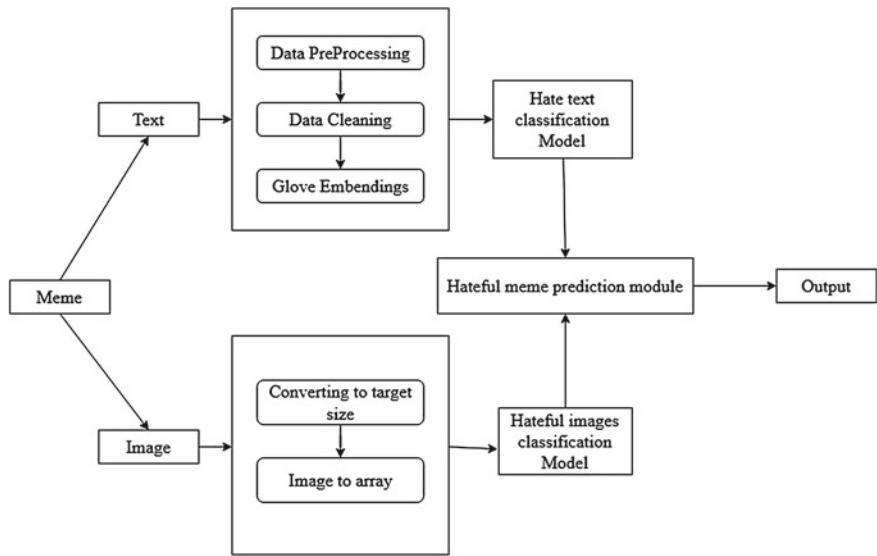


Fig. 4 Modular diagram

wherein the estimation of a word inside the vector space is discovered from text and relies upon the words that incorporate the word when it is utilized. GloVe embeddings and BERT have been used for the same.

Hate Text Classification Model: Experimentation with different architectures has been performed for text classification models—BiLSTM, CNN, Stacked LSTM, etc.

5.2.2 Hateful Image Classification Model

VGG16, Very Deep Convolutional Networks for Large-Scale Image Recognition, architecture has been used as a Hateful Image Classification model to predict whether the image is hateful or not. Features have been extracted based on the image only. Again, the output layer states whether the image is hateful or not.

5.2.3 Hateful Meme Prediction Module

Finally, the Hateful Meme Prediction module involves fusion comprising a concatenation of the outputs received from text model and image model which is then passed through a classifier block giving the output as to whether the meme is hateful or not (Fig. 5).

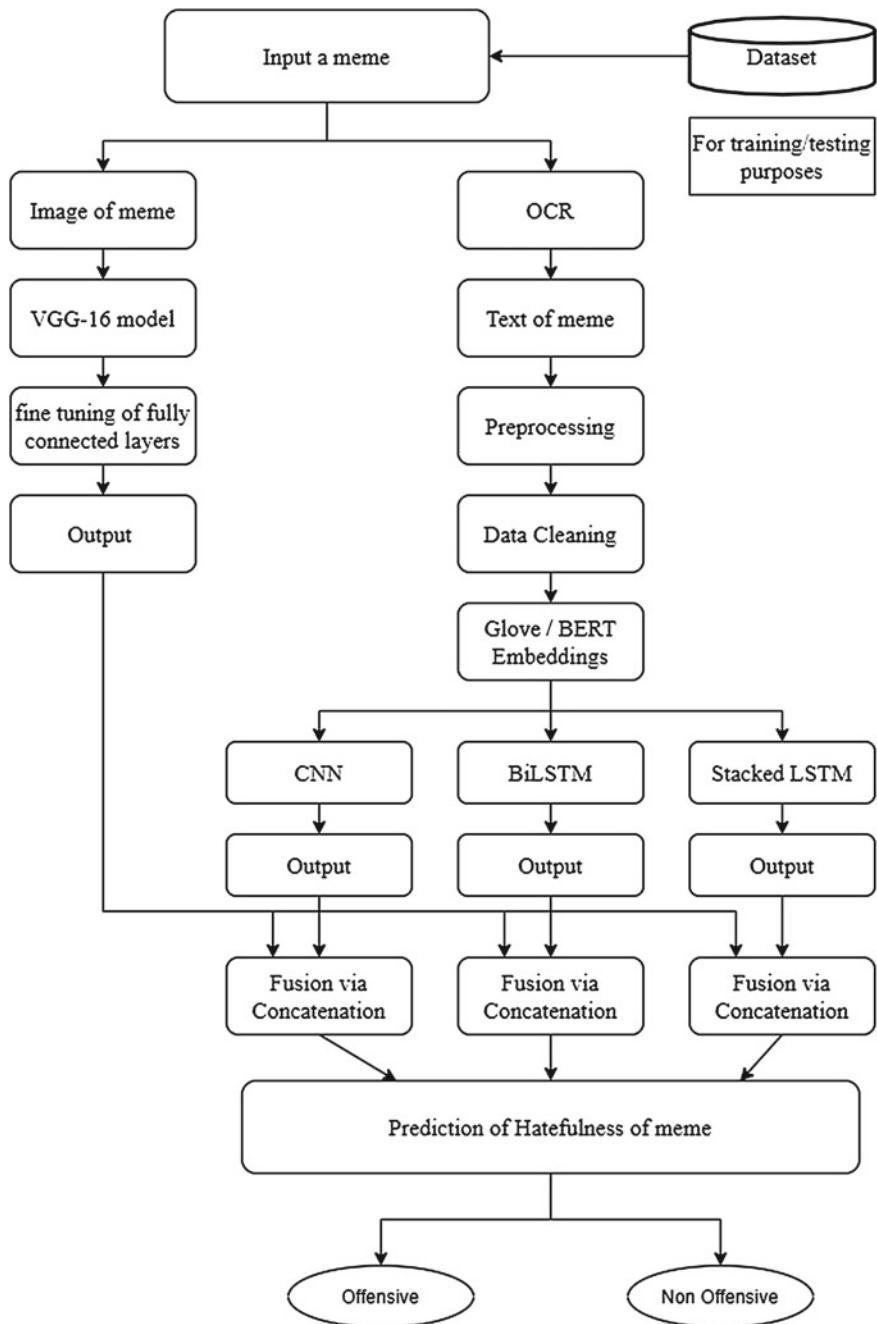


Fig. 5 Flow diagram of a hateful meme classification model

6 Results

The results were compared using F1 scores and the accuracy of a model. Below, the results have been discussed separately for BERT and GloVe (Figs. 6, 7, 8, 9, 10, and 11 and Table 1).

Fig. 6 GloVe BiLSTM + VGG16

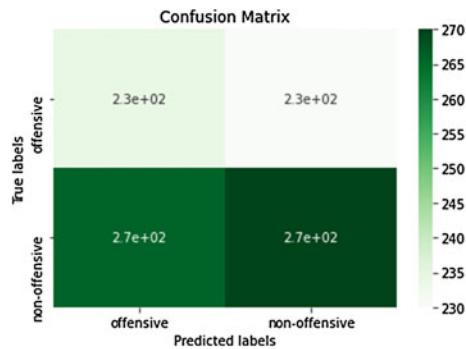


Fig. 7 GloVe CNN + VGG16

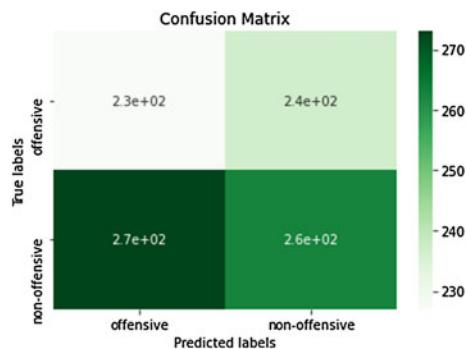


Fig. 8 GloVe Stacked LSTM + VGG16

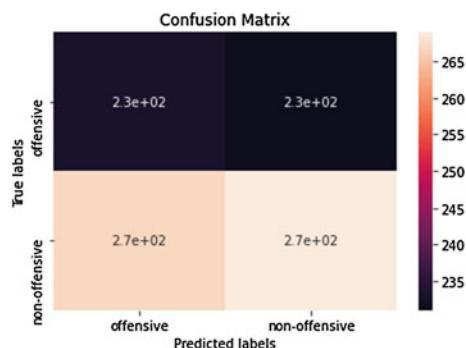


Fig. 9 BERT BiLSTM + VGG16

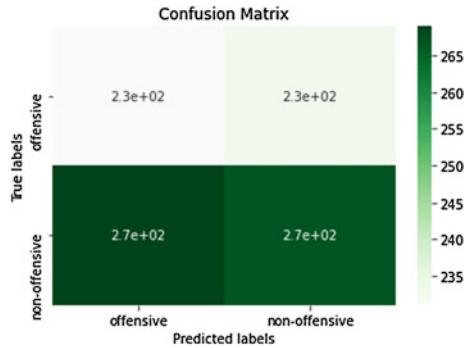


Fig. 10 BERT CNN + VGG16

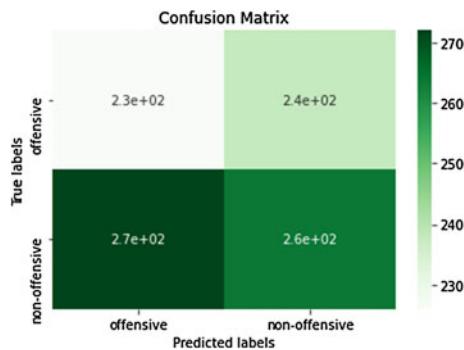
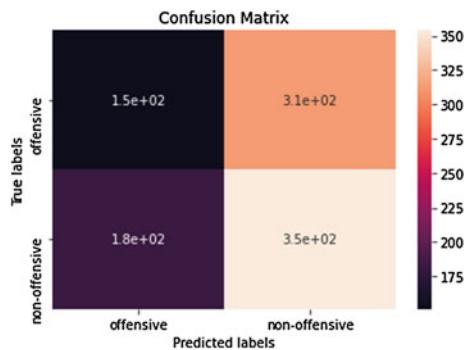


Fig. 11 BERT Stacked LSTM + VGG16



6.1 BERT

Considering only text, BiLSTM outperformed the rest giving an F1 score of 0.494. On combining it with VGG16, BiLSTM and CNN gave approximately the same results with an F1 score of 0.47.

Table 1 F1 score and accuracy for text-only, image-only, and multimodal classifiers

Model	GloVe		Bert	
	F1	ACC	F1	ACC
BILSTM Text	0.495	56.8	0.494	46.4
CNN Text	0.47	35.8	0.487	43.2
Stacked LSTM Text	0.479	53.6	0.456	53.6
BILSTM + VGG16	0.485	49.8	0.479	49.6
CNN + VGG16	0.471	52.6	0.47	54
Stacked LSTM + VGG16	0.470	49.3	0.391	52.9

6.2 GloVe

Here while considering only the text, BiLSTM again outperformed the rest giving an F1 score of 0.495 which is approximately equal to that of BERT. On combining it with the image model, VGG16, BiLSTM still had a boost over others giving an F1 score of 0.485.

7 Conclusion and Future Work

Comparing the results between BERT and GloVe, in our case GloVe beats BERT with a very minor margin, though with an even larger dataset BERT could overtake GloVe. BERT generally outperforms GloVe due to its capability of giving contextual word representations. However, the text of a meme generally contains 2–3 sentences. Thus, our model has approximately the same results for both GloVe and BERT. However, results may vary with the dataset used. In our case, BiLSTM always had a boost over the rest in both GloVe as well as BERT.

Although VGG16 is a powerful CNN model for image classification, comparison of different image classification models can be done using different CNN models to get better results. Also, different fusion techniques can be used and compared with the concatenation technique to improve the results.

References

1. D. Kiela, H. Firooz, A. Mohan, V. Goswami, A. Singh, P. Ringshia, D. Testuggine, The Hateful Memes Challenge: Detecting Hate Speech in Multimodal Memes, 8 June 2020
2. P. Vijayaraghava, H. Larochelle, D. Roy, Interpretable Multi-Modal Hate Speech Detection (2019)
3. A. Bohr, D. Vijay, V. Singh, S.S. Akhtar, M. Shrivastava, A Dataset of Hindi-English Code-Mixed Social Media Text for Hate Speech Detection (2019)

4. T. Connie, M. Al-Shabi, M. Goh, Smart Content Recognition from Images Using a Mixture of Convolutional Neural Networks
5. R. Gomez, J. Gibert, L. Gomez, D. Karatzas, Exploring Hate Speech Detection in Multimodal Publications (2019)
6. F. Yang, X. Peng, G. Ghosh, R. Shilon, H. Ma, E. Moore, G. Predovic, Exploring Deep Multimodal Fusion of Text and Photo for Hate Speech Classification (2019)
7. H. HosseiniMardi, S.A. Mattson, R.I. Rafiq, R. Han, Q. Lv, S. Mishra, Detection of Cyberbullying Incidents on the Instagram Social Network (2015)
8. B.O. Sabat, C.C. Ferrer, X. Giro i Nieto, Hate Speech in Pixels: Detection of Offensive Memes towards Automatic Moderation (2019)
9. P. Rani, S. Suryawanshi, K. Goswami, B.R. Chakravarthi, T. Fransen, J.P. McCrae, A Comparative Study of Different State-of-the-Art Hate Speech Detection Methods for Hindi-English Code-Mixed Data (2020)
10. S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, O. Frieder, Hate speech detection: challenges and solutions (2019)
11. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition (2015)
12. A. Graves, Generating sequences with recurrent neural networks (2013)
13. M. ul Hassan, VGG16—Convolutional Network for Classification and Detection (2018), <https://neurohive.io/en/popular-networks/vgg16/>. Accessed 20 Nov 2018

Drowsiness Detection System Using PPG Sensor's Measured Physiological Parameter



Jyoti Tripathi, Satish Chand, Bijender Kumar, Adrija Ghansiyal, and Anshula Nema

Abstract Drowsiness is generally defined as half-sleep caused mainly by fatigue or lack of complete sleep. In past decade, many researches have been conducted in support of drowsiness detection for drivers with main focus on facial expression recognition, vehicle's movement and physiological signals such as Electroencephalogram (EEG), Electrooculography (EOG), and Electrocardiogram (ECG). This paper presents a comprehensive analysis of the existing methods of drowsiness detection based on physiological parameter-based techniques which can be deployed as a convenient and feasible system. The study also discusses the pros and cons of the diverse methods. Finally, an approach for the detection model is proposed, based on the research findings achieved after an extensive survey. As per the dataset used, the proposed model is able to achieve an accuracy of 90%. Also the proposed model is easy to wear because of its compact size. The system is also fast and starts generating results just after 10 s of its start.

Keywords Drowsiness detection · Electroencephalogram · Electrooculography · Electrocardiogram

1 Introduction

Driving in drowsy state is highly risky that not only may lead to financial loss but also human life. According to Governors Highway Safety Association (GHSA), an US-based NGO, about 5,000 people lost their lives in the year 2015 in U.S. in crashes, due to drowsy driving [1]. The National Highway Traffic Safety Administration (NHTSA), an agency of the U.S. federal government and part of the Department of

J. Tripathi (✉) · B. Kumar
NSUT, Delhi, India

S. Chand
JNU, Delhi, India

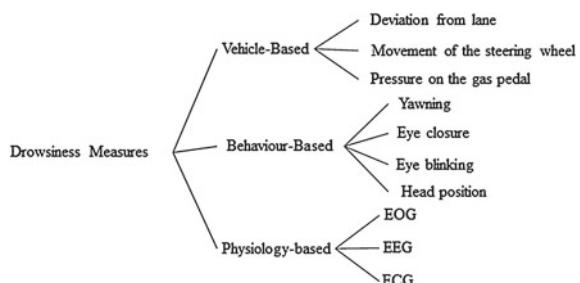
J. Tripathi · A. Ghansiyal · A. Nema
GBPEC, Delhi, India

Transportation, published a report that states about 91,000 police-reported crashes due to drowsy driving, resulting in 50,000 injuries and nearly 800 deaths in US, in 2017 [2]. According to the report by “Ministry of Road Transport & Highways” there were 4,552 accidents reported every year in India, that took lives of thousands of people because of sleepy drivers [3]. The drowsiness not only impacts drivers negatively, but it also leads to impaired cognition and performance, resulting workplace accidents and health consequences in humans. There have been some studies to understand the drowsiness-related functional impairment and the economic expenses for public health, public safety, and productivity.

The drowsiness monitoring in drivers can be done in the form of different measures that may be related to vehicle, behavior or physiology or their different combinations [4]. The vehicle-related parameters include deviations from lane position which is measured by computing the standard deviation of lane position (SDLP), movement of steering wheel which is measured by using steering angle sensor, and pressure on gas pedal that is measured using vehicles brake pedal sensor CL23 [5]. However, the computation of vehicle-based parameters has several drawbacks such as (i) SDLP is entirely dependent on external factors such as road marking, climatic and lighting conditions, (ii) steering wheel movement measurement works only in very limited situations because it can function reliably only in a particular environment and is mainly dependent on the geometric characteristics of the road rather than the kinetic characteristics of vehicle. If there are frequent changes in any or all of these parameters beyond a certain limit, then there are high chances for a driver to feel drowsy. The behavior-related parameters include facial movement, which includes yawning, eye closure, eye blinking, and head position. These parameters are monitored through a camera and the driver is alerted if any of the drowsiness symptoms are detected. The physiology-related parameters, which includes electrooculography (EOG), electroencephalogram (EEG) and electrocardiogram (ECG) are measured by medical sensors [6]. Figure 1 summarizes all these aspects. In comparison to the vehicle-based and behavior-based measurements, the physiological parameters predict drowsiness more accurately because of their strong relationship with the driver fatigue [7], and these parameters are considered in other areas such as industries that employ human beings to control machineries, etc.

As mentioned above, the physiological parameters can predict drowsiness more accurately. An efficient and ergonomically designed hybrid system for detecting

Fig. 1 Different monitoring techniques



drowsiness is proposed that causes no inconvenience to a user as it does not need any heavy heart-monitoring equipment. The system uses Arduino Nano microcontroller and sensor module—MAX30100. The Arduino Nano microcontroller is a pocket friendly component and functionally sound as well. The MAX30100 is small in size and has ultra-low power consumption, thereby having its battery life longer. Further, due to the high signal-to-noise ratio, it provides a good amount of internal noise filtration and can be easily integrated with the smart watches.

The remaining paper is organized as follows. Section 2 presents literature survey. Section 3 introduces the proposed model and its results are discussed in Sect. 4. Finally, Sect. 5 provides the conclusion and future work.

2 Literature Survey

The physiological parameters of a person such as heart-rate, pulse rate, breathing rate, respiratory rate, and body temperature are considered to be highly reliable signals to observe the changes in early stage of drowsiness; whereas, in vehicle or behavioral-based detection of drowsiness, the changes are observable only when the person is in a drowsy state. Sleep and wakefulness cycles are usually regulated by two physiological parameters: circadian cycle (internal clock of body) and homeostatic regulation of sleep (measured through sleep pressure) [8]. Rahim et al. discuss a system that apply infrared sensors and pulse sensors on the steering wheel of the vehicle to measure the beat per minute (BPM) of the driver [9]. Their observation is that as the driver slips to the drowsiness state his BPM decreases rapidly. Barhatte et al. introduce a method for ECG signal classification using the wavelet energy histogram method and support vector machine (SVM) [10]. This method gives good classification accuracy on the MIT-BIH arrhythmia database. Ke et al. discuss a system that acquires the ECG signals using the sensors and computes the power ratio using the hamming window and fast Fourier transform (FFT) [11]. If the power ratio value is above 0.18, the person is supposed to be in an awake state. This system does not require an explicit noise filtering mechanism, but it needs a large database. Babaeian et al. discuss a driver drowsiness detection system that computes the heartbeat rate variation using an advanced logistic regression-based machine learning algorithm, along with noise elimination using the bandpass filters on Arduino [12]. In this work, it is shown that the logistic regression performs better than the Naïve Bayes method. Shahrudin et al. introduce a non-intrusive system that uses a digital Butterworth filter for preprocessing and R-wave features for further processing [13]. The ECG signals have a specific pattern for the heartbeat. That pattern traditionally starts with P-wave that refers to the depolarization of atrium. The other important waves consist of Q, R, and S, which are jointly called QRS complex. The interval between the end of P-wave and the outset of QRS complex is called RR interval. In [13], the state of a person is classified by considering 3 parameters: RR interval, difference of amplitude of R peaks, and the cardioid-based graph analysis, i.e., the Euclidean distance and perimeter of the graphs for normal and drowsy states. It reports that the RR interval

of a drowsy state increases almost by 22% over the normal state and can increase up to 36.33%. Apart from that, the difference of amplitude at R peak for normal to drowsy, normal (current)-to-normal (previous) and drowsy (current) to drowsy (previous) show that the normal amplitude of R peak is always higher than that of the drowsy amplitude, and in a drowsy state, the cardioid graph is much smaller than that of an active state. The cardioid-based graph analysis is applied to only verify the outcomes.

Instead of the exclusive use of the physiological parameters, to detect drowsiness, Gwak et al. conducted a study to identify different alert states of a driver based on hybrid sensing of all three parameters shown in Fig. 1 [14]. They applied machine learning algorithms on these parameter readings which were extracted for 10 s. Their model focused on the ensemble algorithm which presented the accuracy of 82.4% for when physiological indicators were included and an accuracy of 78.7% if they were excluded. Yashwanth et al. discuss a system which uses facial features to detect driver drowsiness by applying Artificial Intelligence-based advanced algorithm [15]. They were able to conclude that the decision tree and neural network classifiers gave better results than linear SVM and LDA and their study also gave a scale for rate of drowsiness on 1–10.

In this study, the authors have focused exclusively on physiological indicators since they yield a better result in detecting drowsiness, as per studies discussed in this section. In the next section, we introduce our proposed system that is based on the ECG mechanism.

3 Proposed Methodology

The proposed system utilizes the heart-rate of an individual that helps identifying the RR interval characteristic of ECG. It takes the beat per minute (BPM) readings and analyzes the RR interval pattern when the transition occurs from awake to drowsy state. This physiological parameter is continuously monitored to differentiate the alertness and drowsiness. The system takes three consecutive BPM values of one second interval using the photoplethysmogram (PPG) sensor. These values of RR interval are compared with the previously recorded values, and if the new values lie between 22 and 36.33% of the previous values, the system declares the person to be in drowsy state; otherwise, in the awake state. This interval is based on the study conducted by Shahrudin et al. [13]. The BPM represents the number of R peaks in a specific time interval. The PPG consists of 2 different modes: transmissive and reflective; and accordingly, the sensors for each mode are designed [8]. Our system uses the reflectance type sensor (MAX30100) that provides the BPM values for every second from which the RR interval duration is calculated as shown in (1) [16].

$$\text{RR interval duration} = \frac{60}{\text{BPM}} \quad (1)$$

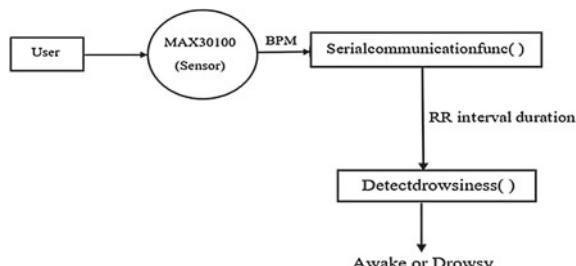
The RR interval duration for each second is recorded and it is compared with the previous value to obtain the final result. When the heart pumps the blood, there is an increase in oxygenated blood, and when the heart relaxes, the volume of the oxygenated blood decreases. The pulse rate is determined by identifying the time between the increase and decrease of the oxygenated blood [17].

It is a known fact that the oxygenated blood absorbs more infrared light and allows to pass more red light, while the deoxygenated blood has greater level of red-light absorption and allows to pass more infrared light. It simply means that the haemoglobin, oxygen-carrying component of red blood cells, is a strong reflector of red light and a strong absorber of infrared light [18]. The MAX30100 reads the absorption levels for both light sources and stores them in a buffer that can be read using the inter-integrated circuit (I2C). Rather than directly processing of the signals received by the sensor, an experimental model is designed that provides the ratio of low frequency (LF) to high frequency (HF) using interpolation and trapezoidal area. When a person goes from the awake state to drowsy state, the heartbeat rate tends to decrease. As a result, the power spectrum of the signal from low frequency (LF) to high frequency (HF) is obtained, which is used as a key parameter to determine whether a person is drowsy state or not. The decreasing trend in the ratio (LF/HF) that tells a person falls asleep for consecutive 3 counts, and it occurs from the increase in power in HF range or the decrease in power in LF range. The functional and working models of the proposed system are shown in Fig. 2 and Fig. 3, respectively.

A. Drowsy-Awake Decision Algorithm

The proposed algorithm waits for 10 s before starting the computation because of two reasons: (1) It has been observed through experimentation, that first 2 readings acquired from the sensor are 0.0 bpm, which is not correct (2) Each RR interval reading requires three more readings apart from itself to compare from. Hence, if there is no waiting time, the system does not have any value to compare with and there will be no output acquired at the end.

Fig. 2 Functional model of proposed system



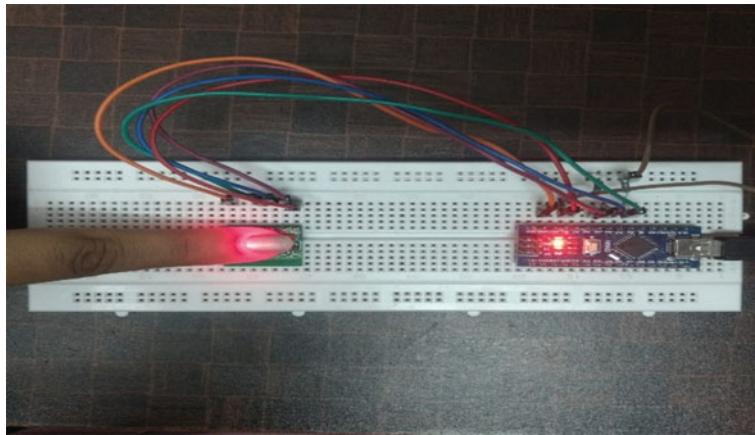
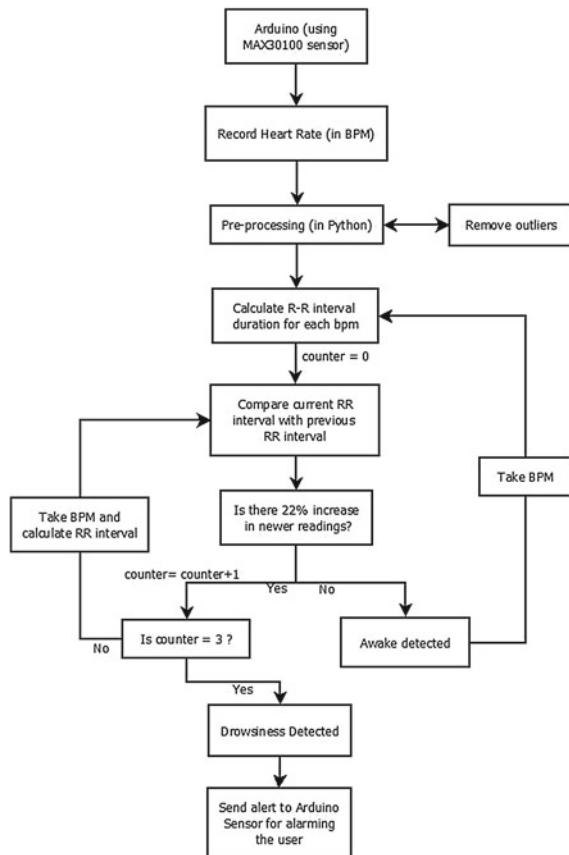


Fig. 3 Working model of proposed system

Input: BPM reading from the PPG sensor.
 Output: Drowsy or awake state.
 Begin
 Align Fingertip/wrist on the sensor incorporated with
 Arduino Nano microcontroller.
 i= 0;
 While(true)
 a. Read heart-rate (bpm) from sensor every second.
 b. i = i+1
 c. Calculate RR interval duration from bpm and store in
 array A[i] = RR interval duration.
 d. If time lapsed is more than 10 seconds OR array
 length of RR interval duration is more than 10, then:
 i. set counter = 0
 ii. If $1.22 * A[i-1] \leq A[i] \leq 1.36 * A[i-1]$
 • Counter++;
 • if counter=3,
 print drowsy state and exit;
 else
 print awake state
 read next bpm reading from sensor,
 calculate RR interval and go to (c.)
 e. Else goto (a.)

Figure 4 shows data flow diagram of the proposed system.
 We now present results and discussions in the next section.

Fig. 4 Data flow diagram of proposed system



4 Results and Discussions

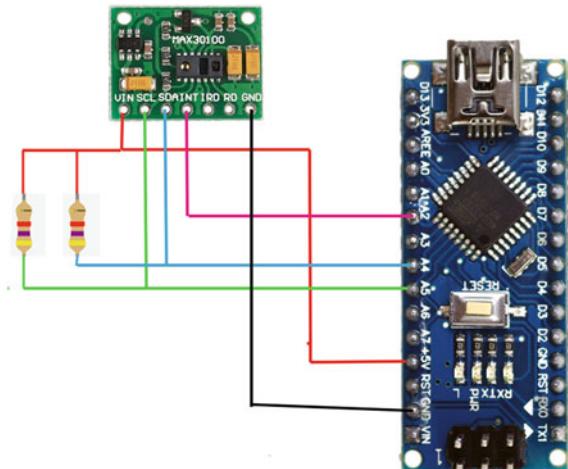
The implementation of the proposed architecture considers the interaction of PPG sensor, Arduino board, and Python module that enable serial communication between the hardware and software. The system consists of the following hardware components: breadboard, Arduino Nano, MAX30100 sensor, jumper cables, and a cable for connecting Arduino board to computer. The circuit connections between Arduino Nano and MAX30100 are summarized in Table 1. In addition to these connections, the sensor module needs be equipped with pins and configured beforehand, to avoid any burnouts or short-circuiting, as shown in Fig. 5. Pull up the resistors to A4 and A5 pins to ensure the complete functioning of the sensor since it works better with 3.3 V supply.

The software setup includes the use of open-source software, Arduino IDE for uploading the module programming into the Arduino microcontroller. The processing of data is done using the Python programming language that involves removing

Table 1 Jumper cable connections between Arduino and MAX30100 sensor

Arduino pin	MAX30100 pin
5V	VIN
A5	SCL
A4	SDA
A2	INT
GND	GND

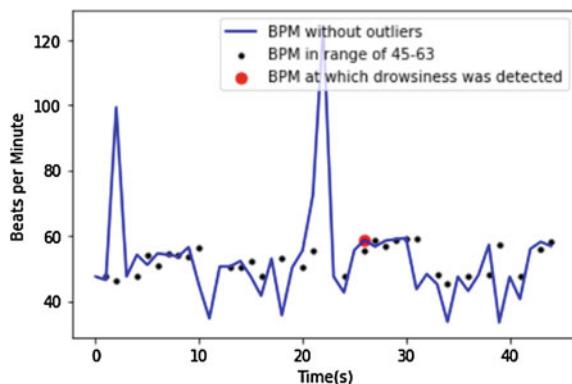
Fig. 5 Wired connection between Arduino Nano and MAX30100



outliers and calculating the heart-rate of a person, for detecting the drowsiness state. We use *interp1d* function from Scipy.interpolate module [19] to interpolate the signal with spline interpolation of third-order to find LF/HF ratio as they can achieve the continuity of the first derivative, but not that of the second derivative. For finding the discrete fast Fourier transform and trapezoidal area of the curve, the NumPy module is used [20]. For Arduino, Wire library is used to communicate with I₂C devices, along with MAX30100 pulse oximeter library to assist in utilizing all features of the sensor [21].

The dataset used in this experiment has been obtained from the heart-rate readings of 10 individuals that is stored in CSV (comma separated values) format in a file. These 10 individuals had been awake for 20 h and were asked to perform their daily activities without any alterations. The BPM readings were recorded when the person was provided with a comfortable and relaxing atmosphere to sleep in. This has been done to get the optimum readings for detecting drowsiness. The heart-rate readings were acquired from the working model that was setup as mentioned above. This dataset consists of 10 readings of different individuals with no specific time restriction in each reading. The reason for this unrestricted behavior is that the drowsiness is not a voluntary activity and one needs be continuously monitored to detect the pattern in his heart-rate over time.

Fig. 6 Comparison of results our proposed system (red dot) and model [9] (black dots). Black dots represent the range of function (45–63 bpm for women) and red dot represents point of drowsiness detection (58.72 bpm)



A preliminary experiment has been conducted using readings from the self-developed dataset on our proposed system and compared with the Rahim et al.'s system [9]. The results have been shown in Fig. 6. As can be seen from this figure, there are many instances when the user is declared drowsy by the Rahim et al.'s model (shown by black dots), which is undesirable and imprecise. This is because it follows a generalized model for every person and categorizes the drowsiness based on the range in which the BPM readings lie. Due to this imprecise and unreliable output, the proposed model has incorporated the model [13]. Our proposed model has provided an exact point in graph when the drowsiness state was detected in the individual (shown in red dot). It provides an improvement in terms of accuracy and precision as compared to the systems [13] and it has generated correct results for 9 out of 10 samples.

The proposed model has further been tested on 2 datasets: stress dataset [22], which is used during initial stages of the model development, and the dataset of 10 individuals that has heart-rate values extracted from the PPG sensor in real time. The proposed system does not require any training dataset. Apart from denoising by the hardware components, the general outliers like BPM value of 0.0 have also been removed from the data that has been replaced by the mean value. The effect of data preprocessing has been shown in Fig. 7 and Fig. 8 that represent the heart-rate of an individual, which are for “data1” and “data2” of the self-generated dataset, respectively. Further, Table 2 shows the detailed analysis of data based on various parameters.

Table 3 shows the results of the proposed systems and the Ke et al.'s model. The proposed model resulted in 9/10 correct responses because in “data9”, the actual output mismatches with that of our model's prediction, whereas the Ke et al.'s model provides the correct output for 5 instances.

During implementation of the proposed system, it has been observed that the accuracy rate of detecting the sleep state is almost inversely proportional to the time the system takes to alert the person. Though there are numerous ways to detect QRS complex, yet the most efficient one is done by using a bandpass filter to denoise the signal and then applying a series of Fourier transforms on it to get the power spectral

Fig. 7 Testing data 1 before and after outlier (BPM = 0.0) removal

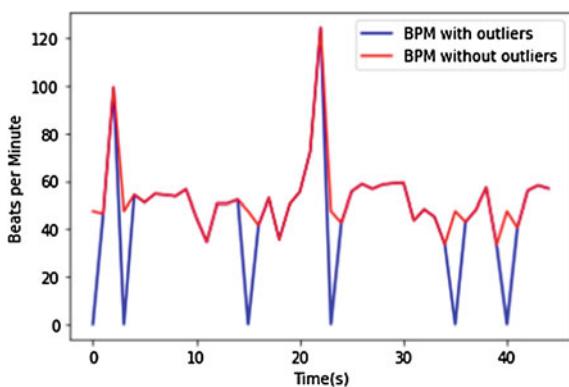


Fig. 8 Testing data 2 before and after outlier (BPM = 0.0) removal

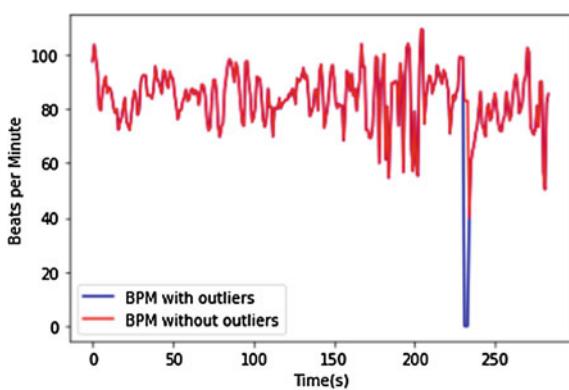


Table 2 Description of the test data 1 and 2: count is the number of readings in a data file with its mean, standard deviation (STD), minimum value (Min) and maximum value (Max)

Parameters	Test data 1	Test data 2
Count	45.00	284.00
Mean	52.75	83.90
STD	15.28	10.14
Min	33.37	40.04
Max	124.35	109.28

density. It however consumes time in the preprocessing especially during real-time analysis. So, it has also been observed that for alerting the subject promptly, some mechanism is needed that can do the preprocessing of the signal efficiently, which has been taken into consideration.

In the analysis of ECG signal, it has also been observed that the R peaks and the interval between them (RR interval) finds significantly as it is a time-domain feature of ECG and assists in the study of heart-rate variability too. The proposed system

Table 3 Comparison of existing model [11] and proposed model

Dataset	Actual output	Ke et al.'s model	Proposed model
Data1	Drowsy	Awake	Drowsy
Data2	Drowsy	Drowsy	Drowsy
Data3	Awake	Drowsy	Awake
Data4	Drowsy	Awake	Drowsy
Data5	Drowsy	Drowsy	Drowsy
Data6	Awake	Drowsy	Awake
Data7	Drowsy	Drowsy	Drowsy
Data8	Awake	Awake	Awake
Data9	Awake	Awake	Drowsy
Data10	Drowsy	Awake	Drowsy

thus takes advantage of the relation between the beats per minute and the RR interval duration to further process the BPM readings into RR interval durations.

5 Conclusion and Future Work

In this paper, we have discussed a system for determining whether a person is in drowsy state or normal state. The proposed system is cost-effective, compact, and has an inbuilt noise filtration mechanism with automatic removal of outliers while processing the data. After 10 s of start/switch on the system starts detecting whether a person is awake or drowsy. While studying and deployment of this model, it has been observed that the accuracy and latency in results are inversely proportional to each other which is an important finding. Our model has some limitations in terms of the hardware used. The module starts getting heated up after 12 min of continuous working. If it is used more than 3 h, it may cause the sensor to report inaccurate measurements. The sensor module must also be placed correctly, with a stable position, because the position and movement of fingers dramatically affects the registered readings. This problem can be addressed by using the MAX86510 sensor, which is costlier than the MAX30100 sensor. But it provides better noise filtration and longer durability. This system can be integrated into a handheld device that could be easily worn by a person in day-to-day life and hence comfortable to wear. The communication with the device to control its power switch and observing the recorded reading for future reference can be assisted with the help of Bluetooth device integration, thus providing wireless connectivity.

References

1. Drivers are Falling Asleep Behind the Wheel. National Safety Council (2019), <https://www.nsc.org/road-safety/safety-topics/fatigued-driving>. Accessed 03 Dec 2019
2. Drowsy Driving. National Highway Traffic Safety Administration (2019), <https://www.safercar.gov/risky-driving/drowsy-driving>. Accessed 03 Dec 2019
3. Road Accidents in India—2016. Government of India (2016)
4. A. Sahayadhas, K. Sundaraj, M. Murugappan, Detecting driver drowsiness based on sensors: a review. Sensors (Basel) **12**(12), 16937–16953 (2012)
5. M. Chciuk, J. Pavlovkin, P. Bachman, Measurement and analysis of pressure forces on pedals at driver's workplace. Pomiary Automat. Robot. 534–537 (2013)
6. M. Ramzan, H.U. Khan, S.M. Awan, A. Ismail, M. Ilyas, A. Mahmood, A survey on state-of-the-art drowsiness detection techniques. IEEE Access **7**, 61904–61919 (2019)
7. M. Awais, N. Badruddin, M. Drieberg, A hybrid approach to detect driver drowsiness. Sensors (2017)
8. F. Bourghelle, Development of an automatic drowsiness monitoring system using the electrocardiogram (2016), <http://hdl.handle.net/2268.2/1451>
9. H.A. Rahim, A. Dalimi, H. Jaafar, Detecting drowsy driver using pulse sensor. J. Teknol. **73** (2015)
10. A.S. Barhatte, R. Ghongade, A.S. Thakare, QRS complex detection and arrhythmia classification using SVM, in *Communication, Control and Intelligent Systems (CCIS)* (2015), pp. 239–243
11. K.W. Ke, M.R. Zulman, H.T. Wu, Y.F. Huang, J. Thiagarajan, Drowsiness detection system using heartbeat rate in android-based handheld devices, in *2016 First International Conference on Multimedia and Image Processing (ICMIP)* (2016)
12. M. Babaeian, N. Bhardwaj, B. Esquivel, M. Mozumdar, Real time driver drowsiness detection using a logistic-regression-based machine learning algorithm, in *IEEE Green Energy and Systems Conference (IGSEC)* (2016)
13. N. Shahrudin, K. Sidek, A. Ismail, Development of a driver drowsiness monitoring system using electrocardiogram. J. Telecommun. Electron. Comput. Eng. **10**, 11–15 (2018)
14. J. Gwak, A. Hirao, M. Shino, An investigation of early detection of driver drowsiness using ensemble machine learning based on hybrid sensing. Appl. Sci. **10** (2020)
15. C. Yashwanth, J.S. Kirar, Driver's drowsiness detection, in *TENCON 2019—2019 IEEE Region 10 Conference (TENCON)* (2019)
16. www.meddean.luc.edu/lumen/meded/medicine/skills/ekg/les1prnt.htm. Accessed 17 Feb 2020
17. P. Celka, N. Granqvist, H. Schwabl, Traditional Tibetan pulse reading in the digital era, vol. 3 (2019)
18. J. Moraes, M. Id, M.X. Rocha, G. Vasconcelos, J.E. Vasconcelos Filho, V. Hugo, V. Albuquerque, A. Alexandria, Advances in photoplethysmography signal analysis for biomedical applications. Sensors **18** (2018)
19. `scipy.interpolate.interp1d`. SciPy.org (2020), <https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.interp1d.html>. Accessed 19 Apr 2020
20. `numpy.fft.fftfreq`. SciPy.org (2020), <https://docs.scipy.org/doc/numpy/reference/generated/numpy.fft.fftfreq.html>. Accessed 19 Apr 2020
21. O. Intersecans, Arduino-MAX30100. GitHub (2018), <https://github.com/oxullo/Arduino-MAX30100>
22. J. Healey, R. Picard, Stress recognition in automobile drivers. IEEE Trans. Intell. Transport. Syst. (2005), <https://physionet.org/content/drivedb/1.0.0/>. Accessed 19 Feb 2020

Deep Ensemble Technique for Short-Term Load Forecasting Using Smart Meter Data



A. L. Amutha, R. Annie Uthra, J. Preetha Roselyn, and R. Golda Brunet

Abstract There has been a growing interest in using the smart meter data for accurate and effective energy management with the availability of smart meter data sourced from homes. Personal load profiles are more complex and hence difficult to forecast compared with aggregate loads. To handle the above tasks, a deep ensemble network is proposed using Spike Neural Network and Long Short-Term Memory to accurately model these predictions. This system utilizes the profiles of different users that are incorporated into the task's understanding. The consumer profiles are subjected to anomaly detection using Spike Neural Network, which detected the anomalies in the data in an online fashion, then the data without anomalies are fed into the Long Short-Term Memory neural framework. The experimental results of the proposed method proved to be superior when compared with the result of Spike Neural Network and Long Short-Term Memory model separately.

Keywords Deep ensemble technique · Long short-term memory · Spike neural network · Load forecasting

1 Introduction

Smart Grid is an integrated, broadly distributed energy supply network that is characterized by two-way electricity and data flow and would be able to monitor anything from power plants to individual appliances to consumer preferences. It integrates the

A. L. Amutha · R. A. Uthra (✉) · J. P. Roselyn
SRM Institute of Science and Technology, Chennai, Tamil Nadu 603203, India
e-mail: annie@srmist.edu.in

A. L. Amutha
e-mail: amuthaa1@srmist.edu.in

J. P. Roselyn
e-mail: preethaj@srmist.edu.in

R. G. Brunet
Government College of Engineering, Salem, Tamil Nadu 636011, India

merits of distributed computing and communications into the grid to provide the data and to balance the demand and supply at the system level. Power has traditionally been distributed to customers in a unidirectional fashion by passive components. With an ever more complex generation and load network, it is imperative that advanced sensor technology such as phasor measurement unit, advanced metering infrastructure, and smart meter are used to constantly observe the electrical conditions of the transmission and distribution network. It allows customers to change energy consumption habits, including the timing and level of demand for electricity. It would increase the distributed generation possibilities, taking the generation of power closer to those it serves. In the Indian power market, the government of India has accelerated the implementation of smart grid technologies. In different states of the nation, 14 smart grid pilot projects have been carried out. Advanced Metering Networks (AMI), Peak Load Management (PLM), Outage Management Systems (OMS), Power Quality Management (PQM), and Distributed Generation (DG) are among the proposed functionalities in these ventures. Load forecasting is a commonly used technique, designed to predict the energy demands of a specific energy grid. The accuracy of a forecast holds great significance for the peak load management functionality of smart grid. It is commonly found to be of two types: long term and short term. Long-term load forecasting (LTLF) typically covers forecast time periods which exceed a year in duration. The load forecasts provided are peak and valley loads for a time period ranging from a week to a month, which are the key to expanding power generation, transmission, and distribution systems. Short-term load forecasting (STLF) covers forecasts over a set period of time from approximately a month to less than a year [1]. For the purpose of load forecasting, various methods have been used such as neural networks, regression models, time series models, statistical algorithms, and many more. But the huge volume and fast-moving nature of smart meter data are posing a great challenge for conventional data mining and machine learning techniques to be applied. So deep neural network architecture is used in this paper for short-term load forecasting.

The paper is structured as follows: in Sect. 2, related works are discussed, deep ensemble technique for load forecasting is elaborated in Sect. 3, experimental results are showcased in Sect. 4, and conclusion is given in Sect. 5.

2 Related Works

A great deal of work has been done in the field of load forecasting to address the problems that the field generates, such as low accuracy over seasons and the lack of specialized consumer profiles for individual geographic areas. In [2], a novel deep learning ensemble was used for the purpose of load forecasting in smart grids and was tested using the smart meter dataset obtained from the Irish Commission for Energy Regulation. The merits were that it was a scalable and flexible forecasting model with high accuracy and good generalization capability. The shortcoming was the large computational cost due to LSTM density. In [3], a Quantile Regression

Neural Network was used for probabilistic load forecasting and was tested on the GEFCom2014 dataset. The merits were that it was a stable and computationally efficient model and also that it produced accurate forecasts. The shortcoming was that it was prone to overfitting. In [4], a kernel-based multi-task learning model was employed for electricity demand forecasting and was tested using the smart meter dataset obtained from the Irish Commission for Energy Regulation. In [5], a Novel Neural Network was used for the purpose of short-term load forecasting and was tested using the dataset of North American Utility. The benefits were its high accuracy predictions and good generalization capability. The demerits were that the Monte Carlo dropout could severely impact accuracy due to the dataset being unpredictable. In [6], a short-term forecasting solution used the idea of applying sister forecasts for load data. In [7], a deep CNN with wavelet decomposition was used for short-term forecasting, tested over wind farm data; the merits of this system were, it gave mostly accurate predictions and also had the ability to be generalized over a period of time. The demerits of the model were that new data tends to skew the results for a while. In [8], a Gaussian process quantile regression-based NN was proposed for short-term load forecasting. Its merits were that it could set an upper and lower bound on the predictions. The demerits of such a model were its heavy computational cost due to the Markov Chain Monte Carlo algorithm and the large data samples required for accurate results [9]. In [10], an LSTM RNN model was proposed for electrical load forecasting. The paper proposed 4 models for comparison using 2 evaluation criteria RMSE and MAPE. The LSTM and SARIMA were more accurate than the implemented NARX and support vector machine and that the SARIMA algorithm was more accurate for upwards trending datasets with defined seasonality. In [11], a novel Load Forecasting (LF) methodology was proposed by utilizing data mining strategies. The proposed LF methodology utilized new anomaly rejection and feature selection systems apart from the new technique for load assessment. An extremely short-term load forecasting by utilizing wavelet neural networks with data pre-filtering was proposed in [12]. Moving forecasts was finished utilizing 12 committed wavelet neural networks. Mathematical testing shows the impacts of data pre-separating and the exactness of wavelet neural networks dependent on a dataset from ISO New England. In [13], a correlation of artificial neural networks (ANN) and support vector machines (SVM) for short-term load forecasting utilizing different load types was proposed. Electricity price forecasting utilizing ANN was proposed in [14]. The fundamental thought was to utilize history and other assessed factors to predict the costs and amounts. Consensus-based combining methods for classifier ensembles are proposed in [15]. It is mentioned that the effectiveness of the approach is evaluated by comparing it with majority voting, product and average methods. Efficient classifier ensemble methods can be found in [16–18].

3 Electrical Load Forecasting Using Deep Ensemble Technique

The proposed model uses Spike Neural Network (SNN) and the Long Short-Term Memory model (LSTM) to predict the future load. The SNN is used to remove the anomalies in the dataset and after removing the anomalies, the data is used to predict the future load using the LSTM model. Figure 1 shows the proposed deep ensemble technique for load forecasting.

3.1 Spike Neural Network

The term “spiking neural network” (SNN) used in this work encompasses any form of neural network that also integrates the notion of time in addition to the neuronal and synaptic states. Neural coding refers to the scheme by which data is denoted by spike trains. The ways of encoding real values comprise temporal encoding, rate encoding, and population encoding [19]. In [20], a temporal neural coding scheme applicable to sequence modeling and forecasting tasks is projected to take advantage of the related temporal events hidden within the sequence. It includes inter-spike interval or time to next spike and probability-based encoding. It uses the precise time of spikes to feed into the network continually. Firing time series is used to represent spiking neuron input and output called spike train. When the neurons fire a pulse, we name it as one firing time. A spiking neuron’s potential is represented via a dynamic (active) variable and acts as an incorporator of the arriving spikes.

Contribution of newer spikes is more than the older spikes toward net action potential. If the incoming spike’s integrated total is bigger than a defined value, then the spike is fired by neuron. This is the reason to say SNN is a dynamic system that is a vital property for online load forecasting. While input neuron ‘ a ’ gets a collection of

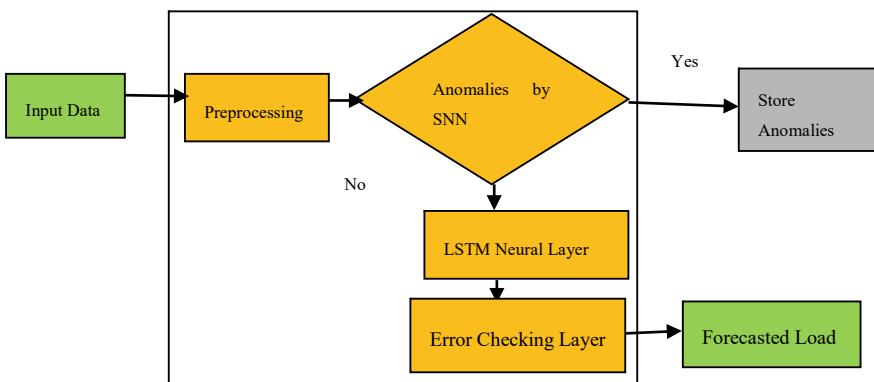


Fig. 1 Deep ensemble technique for load forecasting

spikes having firing times t_a between any 2 neurons ‘ a ’ and ‘ f ’, a neuron’s membrane potential surpasses its defined value (τ) if and only if the output neurons’ spike is fired. The internal state variable and the various input spikes are represented by using the SRM. $x_f(t)$, the state variable, given by Eq. (1) is represented by the response function of spike $\varepsilon(t)$ subjective to synaptic efficacy.

$$x_f(t) = \sum_{k=1}^m Y_a^k W_{af}^k \quad (1)$$

where W_{af}^k represents the weight of sub-connection k, m is the count of synaptic terminals between any two neurons of successive layers, and Y given by Eq. (2) provides the synaptic terminal of all the un-weighted spike contribution

$$Y_a^k(t) = \varepsilon(t - t_a - d^k) \quad (2)$$

where d^k represents the delay of two nodes in the k^{th} synaptic terminal and $e(t)$ defines the spike response function in the method of shaping the post-synaptic potentials (PSP) of a neuron, as given by Eq. (3).

$$\varepsilon(t) = \begin{cases} \frac{t}{\tau} e^{1-\frac{t}{\tau}} & t \geq 0; 0t < 0 \end{cases} \quad (3)$$

where membrane potential decay time constant is represented by τ .

Spike Propagation Algorithm.

1. Make the dataset standardized between 0 and 1.
2. Small random values are assigned with various weights and the parameters τ and α are initialized.
3. Repeat steps 4–7 for the specified number of iterations. The numbers of neurons are represented by the network indices. The network input is applied with the input vector to t .
4. Compute the internal state variable of the hidden-layer neurons using Eq. (4) as

$$x_h(t) = \sum_{i=1}^p \sum_{k=1}^m Y_i^k W_{ih}^k \quad (4)$$

5. Compute the network output using Eq. (5) as

$$x_j(t) = \sum_{h=1}^q \sum_{k=1}^m Y_h^k W_{hj}^k \quad (5)$$

6. Compute the error using Eq. (6) as

$$E = 0.5 \times \sum_{j=1}^n (t_j^a - t_j^d)^2 \quad (6)$$

where t_j^a is the actual output spike, t_j^d is the desired output spike, and n is the neurons in the output layer.

7. Adjust the weight of the networks in such a way that minimizes the error (E). The change in weights from hidden layer to output layer is given by Eq. (7) as

$$\Delta W_{hj}^k = -\alpha \delta_j Y_{hj}^k(t_j^a) \quad (7)$$

The change in weight from input (i) to hidden layer (h) is given by Eq. (8) as

$$\Delta W_{ih}^k = -\alpha \delta_h Y_{ih}^k(t_h^a) \quad (8)$$

where

$$\delta_j = \frac{t_j^d - t_j^a}{\sum_{k,h} W_{hj}^k \left(\frac{\partial Y_{hj}^k(t_j^a)}{\partial t_j^a} \right)} \quad (9)$$

where

$$\delta_h = \frac{\sum_{j=1}^n \delta_j \sum_k W_{hj}^k \left(\frac{\partial Y_{hj}^k(t_j^a)}{\partial t_j^a} \right)}{\sum_{k,i} W_{ih}^k \left(\frac{\partial Y_{ih}^k(t_h^a)}{\partial t_h^a} \right)} \quad (10)$$

Long Short-Term Memory Model.

LSTM neural networks or long short-term memory networks were created to solve the problem of vanishing error; they do this by storing relevant information to the cell state for as long as needed. LSTMs have been shown to outperform most, if not all, traditional algorithms in accuracy and ease of prediction. Based on the survey conducted on neural networks in STFL, a novel architecture to perform load forecasting is proposed. The proposal implements the concept of LSTM referenced above combined with a scoring mechanism. The proposed system consists of the following: the input layer to remove outliers and prime data for processing. The input layer will then consist of a denoiser to remove anomalous data values and a flag function to log anomalies. The purpose of this flagging and storing technique is to preserve those continuous anomalies for future reference. The data parser will separate the long continuous data into discrete packets of a set amount. This will allow us to simulate live data streaming and adapt the model to real-time changes and conditions. The LSTM neural layer to function as a predictor consists of an LSTM. LSTM is chosen for this purpose as it is able to model long-term dependencies which are very useful in the context of making predictions and outperform most other traditional

forecasting algorithms in the challenging short-term load forecasting scenario [10]. The LSTM takes the primed data from the input layer and processes it to output load value. The error checking layer is used to validate the output. This layer runs Mean Squared Error (MSE) as its primary algorithm. Predicted load data is accepted from the LSTM layer and then verified by checking with the actual values from the test dataset.

4 Experimental Results

The dataset used in this research is the electricity consumption data (December 2016–June 2018) collected inside the IIT Bombay campus [21], and the building consists of 3BHK apartments with a count of more than 60 units, each fitted with a smart meter, with a sampling time of 5 s logging data. The data used is downsampled at 1 h granularity. For experiments, we used only one apartment data, and 80% of the data is used for training and 20% is used for testing. The actual and forecasted load for all three phases is shown in Fig. 2. The evaluation metric used is the mean absolute percentage error (MAPE). The comparison of the proposed work with existing work is given in Table 1.

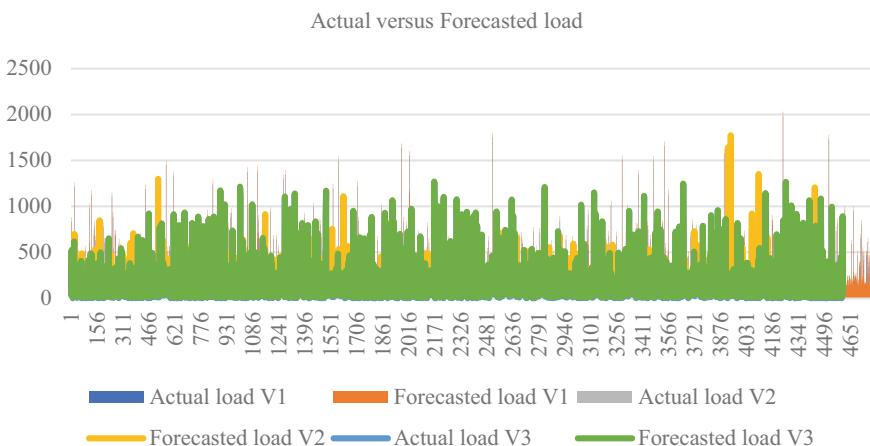


Fig. 2 Actual versus forecasted load for all three phases

Table 1 Comparison of proposed method with existing method

Methods	MAPE (%)
SNN	3.356
LSTM	2.532
Deep ensemble technique	1.453

5 Conclusion

We proposed a deep ensemble technique for short-term load forecasting based on spike neural networks and a long short-term memory model. The spike neural network was used to remove anomalies from the dataset, the LSTM model is used to forecast the load, and the ensemble of these two strategies allows the proposed model to have high accuracy as well as satisfactory generalization capability. The proposed model is capable of forecasting the load in a real-time manner. The dataset we used was obtained from IIT Bombay, and the experimental results were compared with spike neural network and the long short-term memory model. The results proved that after removing the anomalies, the forecasted result was highly accurate than the result without removing the anomalies.

References

1. T. Anwar, B. Sharma, K. Chakraborty, H. Sirohia, Introduction to load forecasting. *Int. J. Pure Appl. Math.* **119**(1), 1527–1538 (2018)
2. Y. Yang, W. Hong, S. Li, Deep ensemble learning based probabilistic load forecasting in smart grids. *Energy* **1**(1), 1–10 (2019)
3. W. Zhang, H. Quan, D. Srinivasan, An improved quantile regression neural network for probabilistic load forecasting. *IEEE Trans. Smart Grid* **10**(4), 4425–4434 (2019)
4. J. Fiot, F. Dinuzzo, Electricity demand forecasting by multi-task learning. *IEEE Trans. Smart Grid* **9**(2), 544–551 (2018)
5. K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, J. He, Short-term load forecasting with deep residual networks. *IEEE Trans. Smart Grid* **10**(4), 3943–3952 (2019)
6. B. Liu, J. Nowotarski, T. Hong, R. Weron, Probabilistic load forecasting via quantile regression averaging on sister forecasts. *IEEE Trans. Smart Grid* **8**(2), 730–737 (2017)
7. H.-z Wang, G.-q Li, G.-b Wang, J.-c Peng, H. Jiang, Y.-t Liu, Deep learning based ensemble approach for probabilistic wind power forecasting. *Appl. Energy* **188**(1), 56–70 (2017)
8. Y. Yang, S. Li, W. Li, Q. Meijun, Power load probability density forecasting using Gaussian process quantile regression. *Appl. Energy* **213**(1), 499–509 (2018)
9. M. De Felice, X. Yao, Notes short-term load forecasting with neural network ensembles: a comparative study. *IEEE Comput. Intell. Mag.* **6**(1), 47–56 (2011)
10. J. Zheng, C. Xu, Z. Zhang, X. Li, Electric load forecasting in smart grid using long-short-term-memory based recurrent neural network (2017)
11. A.I. Saleh, A.H. Rabie, K.M. Abo-Al-Ez, A data mining based load forecasting strategy for smart electrical grids
12. C. Guan, P.B. Luh, L.D. Michel, Y. Wang, P.B. Friedland, Very short-term load forecasting: wavelet neural networks with data pre-filtering. *IEEE Trans. Power Syst.* **28**(1), 30–41 (2012)
13. G. Mitchell, S. Bahadoorsingh, N. Ramsamooj, C. Sharma, A comparison of artificial neural networks and support vector machines for short-term load forecasting using various load types
14. D. Singhal, K.S. Swarup, Electricity price forecasting using artificial neural networks
15. O. Alzubi, A. Jafar, S. Tedmori, H. Rashaideh, O. Almomani, Consensus-based combining method for classifier ensembles. *Int. Arab. J. Inf. Technol.* **15**(1), 76–86 (2018)
16. O.A. Alzubi, J.A. Alzubi, M. Alweshah et al., An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* **32**(1), 16091–16107 (2020)
17. O. Alzubi, Deep learning-based intrusion detection model for industrial wireless sensor networks. *J. Intell. Fuzzy Syst.* (2020)

18. A. Khamparia, A. Singh, D. Anand, D. Gupta, A. Khanna, N.A. Kumar, J. Tan, A novel deep learning based multi-model ensemble methods for prediction of neuromuscular disorders. *Neural Comput. Appl.* (Springer) (2018)
19. V. Sharma, D. Srinivasan, A spiking neural network based on temporal encoding for electricity price time series forecasting in deregulated markets, in *International Conference on Neural Networks* (2010)
20. H.G. Rotstein, F. Nadim, *Neurons and Neural Networks: Computational Models* (Wiley and Sons, Hoboken, 2001)
21. P.M. Mammen, H. Kumar, K. Ramamritham, H. Rashid, Want to reduce energy consumption, whom should we call? in *Proceedings of the Ninth International Conference on Future Energy Systems* (2018)

Performance Analysis of ISOWC Link Considering Different System Parameters



Sanmukh Kaur, Anurupa Lubana, and Anuranjana

Abstract FSO, besides providing all the tremendous technical developments in the arena of distant wireless communications, has also been researched by many scientists and scholars for ISOWC. ISOWC important features are high bandwidth, compact dimensions, comparatively lesser weight and economic feasibility. The objective of this work is not just to explore the ideas of deployment and the advantages of ISOWC (inter-satellite free space optical wireless communication) but to analyze and investigate the performance of the system considering different internal parameters of the system. Here, in this work, we are analyzing and optimizing the different aspects of an optical communication link by selecting feasible combination of wavelength, type of detectors, transmitter and receiver aperture diameter, transmitted input power, the kind of filters and type of modulators.

Keywords FSO · ISOWC · Wavelength · Bit rate · Q-factor

1 Introduction

Optical Communication has been used in many applications from short-range in-door appliances via infrared technology to large-scale terrestrial wireless and guided media using fibre cables providing effective connectivity [1–6]. Apart from these applications, it has also been implemented in extremely large-scale projects, one of which is Inter-Satellite Wireless Optical Communication (ISOWC), which majorly relied on microwave communication technology. There are many reasons to select free space

S. Kaur · A. Lubana (✉) · Anuranjana

Amity School of Engineering and Technology, Amity University, Noida, Uttar Pradesh, India
e-mail: anurupalubana83@gmail.com

Anuranjana
e-mail: aranjana@amity.edu

A. Lubana
Ambedkar Institute of Technology, Delhi, India

optical communication over the traditionally implemented microwave communication when it comes to providing wireless communication channel between two satellites [7]. ISOWC important features are high bandwidth, compact dimensions, comparatively lesser weight and economic feasibility. Here, in this work, we study the effect of wavelength variation and selection of the type of detector along with exploring the impacts of variation of aperture diameter for optimizing the overall performance of the system [8].

In this work, the objective is to design and analyze the performance of inter-satellite link between two satellites with minimum loss and high speed and to get better link availability that covers maximum range by analyzing ISOWC Link with different internal system parameters such as power, range, type of detectors used, wavelength, transmitter and receiver aperture diameter, etc.

2 Literature Review

The signal power loss in OWC systems is less as compared to radio frequency (RF) systems. The regulation and license to frequencies that can be used for satellites communication using RF links are not applicable in case of optical system [3]. Free space optics (FSO) is preferred over other systems as it uses license free spectrum, work at higher bit rates, higher bandwidth, less input power but it is not suitable in cases where atmospheric conditions are poor for communication [4]. FSO system offers many advantages over RF systems as it consumes less power and have higher bandwidth capability along with high bit rate and free license spectrum [7]. FSO system depends upon various internal and external parameters such as transmission bandwidth, input power, wavelength of operation and beam divergence, etc. [8]. The idea of linking LEO (Lower Earth Orbit) satellites and geostationary satellites through optical links is not new and has been implemented and discussed in literature for quite a while now. In [9], author used higher order modulation techniques and compared two different types of detection schemes for an ISOWC link. The ISOWC Link analysis using space and polarization diversity methods have been explored in [10]. Various internal parameters of the system have been varied in [11] to investigate their impact on performance of FSO link. In [12], by varying beam divergence and wavelength, Q-factor at the output of the link has been analyzed at two different bit rates of 10 and 15 Gbps. An ISOWC Link has been proposed at a wavelength of 1550 nm using NRZ modulation techniques at 1 Gbps.

An ISOWC Link has been analyzed between two LEO satellites and it has been reported that the scattering effect can be abridged by using extended wavelength such as 1550 nm [13]. In [14], author evaluated the FSO Link performance by selecting suitable operating wavelength, detector type and pointing errors at transmitter and receiver side. Performance of ISOWC Link has also been evaluated using different modulation formats, at different data rates for LEO and MEO distances in [6]. This work has been divided into five sections, literature survey has been explored in section

one and two, followed by section third and fourth which illustrates the system set up and results, respectively. Finally, conclusion has been described in Sect. 5.

3 Proposed Configuration

The proposed system set up is shown in Fig. 1 and system model in Fig. 2. OptiSystem-16 has been used for designing and simulation the system the model. The system has been provided with input wavelength of 850 nm and power of value 12 dBm. The ISOWC communication system comprises of a transmitter, communication channel and a receiver having a photo-detector (PIN) and a low pass Bessel filter. System design parameters and their corresponding values have been given in Table 1.

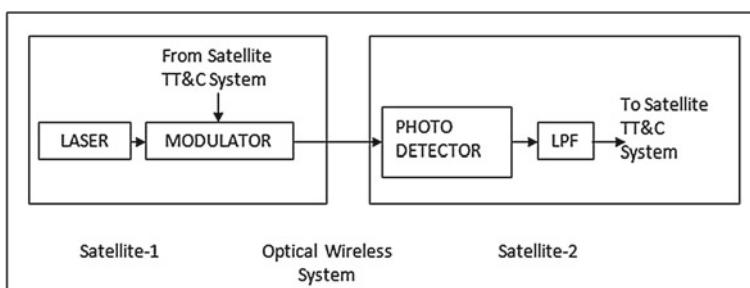


Fig. 1 ISOWC system set up

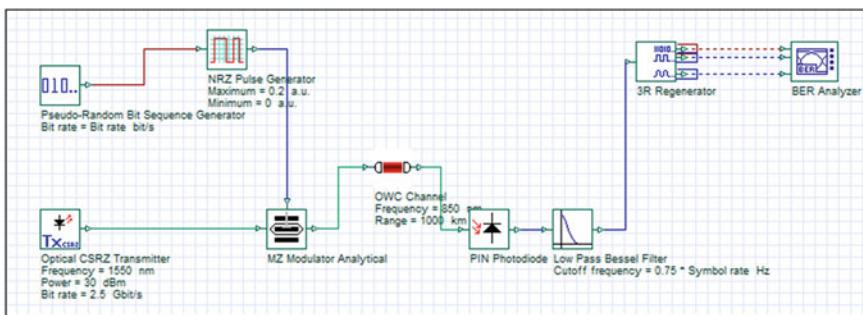


Fig. 2 ISOWC system model

Table 1 ISOWC system specifications

S. no	Parameters	Values
1	Power	12 dBm
2	Bit rate	1 e + 009
3	Wavelength	850 nm
4	Aperture diameter (transmitter)	15 cm
5	Aperture diameter (receiver)	15, 25, 35 cm
6	Range	1000 km
7	Attenuation	0 dB/km
8	Detector type	APD
9	Modulation type	NRZ
10	Additional losses	0 dB

4 Results and Discussion

A. This Optimization of wavelength for ISOWC link between satellites

In the plot shown in Fig. 3, three wavelengths have been analyzed for the performance of ISOWC link. The analysis of 850, 1300 and 1550 nm has been performed in terms of Q-factor for transmission range of 1000 km. As seen in the Fig. 3, it is clear that 850 nm provides better Q-factor in comparison of 1300 and 1550 nm and as it is clear from the figure that by using 850 nm the system provides better range availability than at 1300 and 1550 nm.

B. Optimization of detectors with range

Here in this work, two type of photo detectors have been analyzed and compared in terms of Q-factor with varying transmission range. Figure 4 shows the plot of Q-factor with respect to transmission range in km for PIN and APD detectors. It is evident from the figure that APD has an edge over the PIN detector in terms of achieved Q-factor. With the use of APD detector, the system performs for long range

Fig. 3 Q-factor versus transmission range with varying wavelength

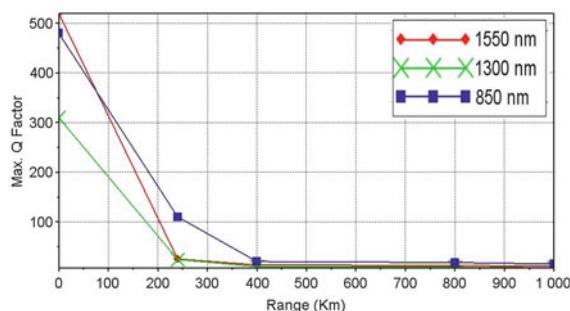
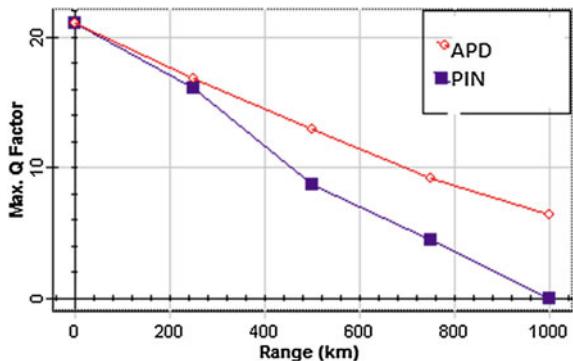


Fig. 4 Q-factor versus range with different detectors



and with PIN detector the system performs for short distance, and thus APD detector with gain 3 has been optimized for further simulations.

C. Receiver aperture diameter optimization

Figure 5 shows that four different values of receiver aperture diameters have been used to evaluate the efficiency of ISOWC link. It is seen in Fig. 5 and Table 2 that, when small aperture diameter of 15 cm is used, lower Q-factor has been achieved and system performs well for smaller distance up to 500 km and while using higher receiver aperture diameter, the better Q-factor is achieved and system performs better for 800–900 km than at lower aperture diameter. It is shown in the Table 2 that Q-factor increases with the increase of receiver aperture diameter. Receiver aperture diameter of 35 cm has been optimized for the further simulations.

D. Optimization of transmitter aperture diameter

Here, transmitter aperture diameter has been analyzed by keeping the optimized values of parameters discussed in previous sections. Figure 6 shows that Q-factor increases linearly with transmitter aperture diameter. The graph below shows that

Fig. 5 Q-factor versus range with varying receiver aperture diameter

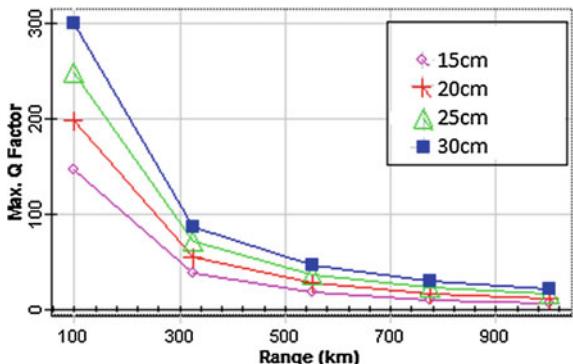
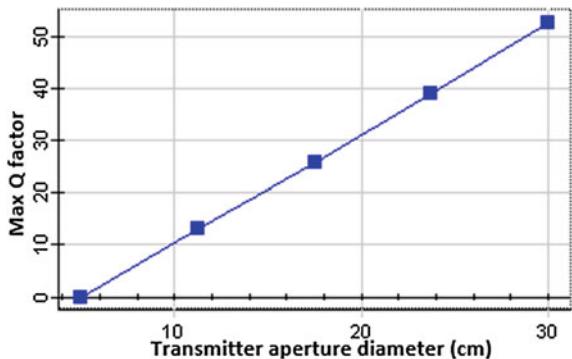


Table 2 Q-factor variation with varying receiver aperture diameter

Receiver aperture diameter (cm)	Distance (km)	Bit rate (Gbps)	Q-factor
15	100	1	146.75
15	1000	1	6.54
20	100	1	197.99
20	1000	1	10.76
25	100	1	247.851
25	1000	1	15.52
35	100	1	298.92
35	1000	1	20.50

Fig. 6 Q-factor versus transmitter aperture diameter with range of 1000 km



when transmitter aperture diameter is 1 cm, then 0 Q-factor is achieved and when there is slightly increase in transmitter aperture diameter, then Q-factor also increases slightly, and when diameter is taken as 30 cm then Q-factor achieved is 52.36. Transmitter aperture diameter of 30 cm has been optimized for the further analysis.

E. Optimization of data rate for ISOWC link

The Q-factor plot with respect to power at three data rates of 5 mbps, 50 mbps, 500 mbps and 5 Gbps has been depicted in Fig. 7. From the plot, it is shown that while using low bit rate, i.e. 5 Mbps the higher Q-factor is achieved and while using a bit rate of 5 Gbps, the lower Q-factor has been observed. As the bit rate increases, Q-factor starts decreasing. The system performs better at 12 dBm as increasing the power beyond this value does not increase the quality of signal significantly for a given data rate.

F. Optimization of filters with varying range

In this section, Fig. 8 shows the maximum Q-factor with respect to transmission range using a Gaussian filter and a Low Pass Bessel filter. From the graph, it is seen that with the use of electrical low pass Bessel filter, better Q-factor is achieved as

Fig. 7 Q-factor versus power considering different data rates

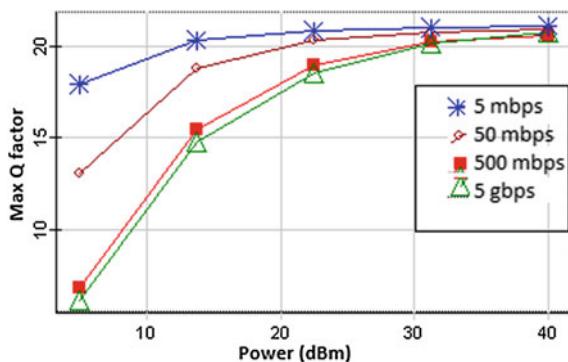
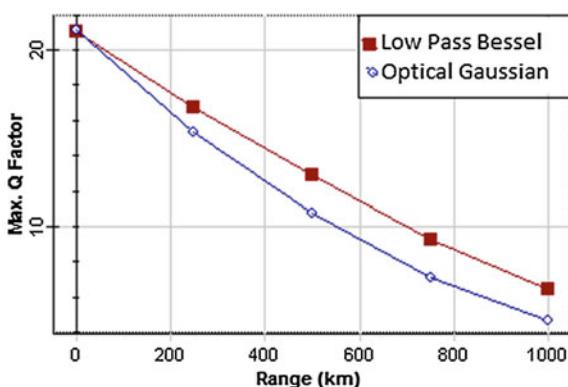


Fig. 8 Q-factor versus range for different filters



compared to Gaussian filter. Low pass Bessel filter has been optimized for the further analysis and variation of Q-factor with range for different filters have been mentioned in Table 3.

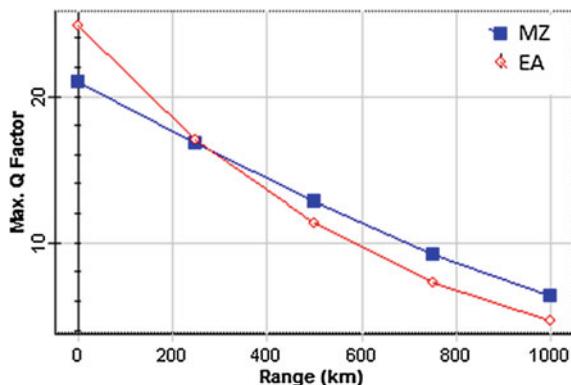
G. Optimization of modulators with varying range

Here in this section, the graph is plotted between range and maximum Q-factor considering different modulators, i.e. Mach-Zender (MZ) and electro-absorption (EA) modulator. MZ modulator is used for controlling the amplitude of an optical

Table 3 Variation of Q-factor with range for different filters

Range (km)	Low pass Bessel filter Q-factor	Gaussian Q-factor
0	20.99	21.10
250	16.75	15.29
500	12.86	10.76
750	9.20	7.12
1000	6.41	4.67

Fig. 9 Q-factor versus range for different modulators



wave and helps in generating optical pulse with tunable pulse width at double clock rate. An EA Modulator modulates the intensity of laser beam via an electric voltage. From Fig. 9, it is observed that MZ modulator performs better than EA modulator in terms of Q-factor as it provides better Q value at longer range than EA modulator.

5 Conclusion

In the present work, various system parameters have been analyzed, simulated and optimized to get better results for determining the link availability of ISOWC link. The system performs better at 12 dBm as increasing the input power beyond this value does not increase the quality of signal significantly for a given data rate. The wavelength at which the link covers extended transmission range is 850 nm. The communication link has been designed for 1000 km of range. Transmitter and receiver aperture diameters have also been analyzed and it is concluded that with the use of 30 cm receiver and transmitter aperture diameter, the system gives better Q-factor and provides better link availability. Better performance has been achieved by employing Low pass Bessel filter as compared to gaussian filter. It is observed that MZ modulator performs better than EA modulator in terms of Q-factor as it provides better Q value at longer range than EA modulator.

References

1. S. Samadi, M.R. Khosravi, J.A. Alzubi, O.A. Alzubi, V.G. Menon, Optimum range of angle tracking radars: a theoretical computing. Int. J. Electr. Comput. Eng. (IJECE) (2019). <https://doi.org/10.11591/ijece.v9i3.pp1765-1772>
2. A. Albalawi, H. Zhu, S. Taccheo, A. Chiasera, M. Ferrari, J. Alzubi, O. Alzubi, Numerical modeling of the impact of pump wavelength on Yb-doped fiber amplifier performance. Opt. Quant. Electron. (2016). <https://doi.org/10.1007/s11082-016-0771-z>

3. H. Kaushal, G. Kaddoum, Optical communication in space: challenges and mitigation techniques, *IEEE Commun. Surv. Tutor.* (2016)
4. Kaushal, V. Jain, S. Kar, Free-space optical channel models, in *Free Space Optical Communication Optical Networks* (Springer (India) Pvt. Ltd., New Delhi, 2017), pp. 41–89
5. Anurupa, S.Kaur,Y. Malhotra, Performance evaluation and comparative study of novel high and flat gain C plus L band Raman plus EYDFA co-doped fiber hybrid optical amplifier with EYDFA only amplifier for 100 channels SD-WDM systems. *Opt. Fiber Technol.* **1**, 53 (2019). <https://doi.org/10.1016/j.yofte.2019.102016>
6. A. Lubana, S. Kaur, Y. Malhotra, Performance enhancement of Raman+EYDFA HOA for UD-WDM system applications. *J. Opt. Commun.* 000010151520200195 (2020). <https://doi.org/10.1515/joc-2020-0195>
7. S. Kaur, A. Kakati, Analysis of free space optics link performance considering the effect of different weather conditions and modulation formats for terrestrial communication. *J. Opt. Commun.* (2018). <https://doi.org/10.1515/joc-2018-0010>
8. S. Kaur, Analysis of inter-satellite free-space optical link performance considering different system parameters. *Opto-Electron. Rev.* **27**, 10–13 (2019). <https://doi.org/10.1016/j.opelre.2018.11.002>
9. W. Rosenkrans, S. Schaefer, Receiver design for optical inter-satellite links based on digital signal processing, in *18th International Conference on Transparent Optical Networks* (ICTON) (2016), pp. 1–4. <https://doi.org/10.1109/ICTON.2016.7550381>
10. S. Pradhan, P.K. Sahu, R.K. Giri, B. Patnaik, Inter-satellite optical wireless communication system design using diversity techniques, in *International Conference on Microwave, Optical and Communication Engineering (ICMOCE)* (2015), pp. 250–253. <https://doi.org/10.1109/ICMOCE.2015.7489738>
11. M. Kaur, Anuranjana, S. Kaur, A. Kesarwani, P.S. Vohra, Analyzing the internal parameters of free space optical communication, in *7th International Conference on Reliability Infocom Technologies and Optimization* (IEEE, 2018)
12. A.H. Hashim, F.D. Mahad, S.M. Idrus, A. Supa, Modeling and performance study of inter-satellite optical wireless communication systems, in *International Conference on Photonics* (IEEE, 2010)
13. V.S. Kiran, V. Kumar, A. Turuk, S. Das, Performance analysis of inter-satellite optical wireless communication. *Int. J. Comput. Netw. Inf. Secur.* **4**, 22–28 (2017)
14. S. Kaur, Anuranjana, R. Goyal, Analysis of terrestrial FSO link performance considering different fog conditions and internal parameters of the system, in *6th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, India (2019), pp. 552–557. <https://doi.org/10.1109/SPIN.2019.8711577>

Video Summarization Using SIFT Features and Niblack's Thresholding



Amol Shinde, Dipti Jadhav, and Swapnil Shinde

Abstract Video Summarization is a technique that aims to generate a short video summary of the given video and helps to understand the context of the video. A lot of research has been done on reducing the redundant images in the video summary and obtaining optimized frames through various techniques. Researchers have also worked on eliminating the redundancy of images and the complexity of the process for achieving these optimized frames. In this paper, we propose an efficient system to perform video summarization using the SIFT features along with feature differencing. The feature differencing applied along with Niblack's threshold on feature vectors helps to eliminate the redundant frames. The proposed technique is tested on the open vide database videos viz, v24 and v37. The precision and recall values calculated for open video library V24 and V37 videos are comparable to the classical video summarization techniques proposed in the literature. Performance analysis done for the proposed technique based on subjective and comparative show comparable results.

Keywords Feature vector differencing · Keyframe extraction · Niblack's threshold · SIFT · Video summarization

1 Introduction

Video Summary aims to develop strategies that extract keyframes from a video based on the video content [1]. There are two types of summarized videos, viz., still keyframes and video skims. Dynamic video summarization is a tedious task, but

A. Shinde (✉) · D. Jadhav · S. Shinde
Department of IT, RAIT, Nerul, Mumbai, India
e-mail: ashinde87@gmail.com

D. Jadhav
e-mail: Dipti.jadhav@rait.ac.in

S. Shinde
e-mail: Swapnil.shinde@rait.ac.in

these dynamic videos can be used to generate meaningful outcomes and summaries. Various image feature extraction techniques have been proposed like Scale Invariant Feature Transform [2] (SIFT), Blob Detection Method, Speeded Up Robust Features (SURF) for feature extraction and image matching. SIFT was proposed by Lowe and SURF by Bay [3]. In this paper, we propose to apply feature differencing after SIFT feature extraction along with Niblack's thresholding [4]. The feature differencing vector helps to reduce redundant frames and gives unique keyframes at the end of the process.

The paper is arranged as follows. The study of the literature survey is covered in the next section, followed by details and a block diagram of the proposed system. Then the later section represents the results and analysis part, and the final section concludes the paper.

2 Literature Survey

Xingteng, et al. [5] proposed a new faster image comparison method; we can use the k-d tree building method and use the advanced BBF technique to insert a straight algorithm, and then compare the performance of the revised method by testing.

The authors in [6] proposed a solution to improve matching accuracy with the help of three kinds of strategies like Lowe's optimization algorithm, an algorithm for measuring the size of the definition and how to do it directly. Test results show that the Lowe application method has very high accuracy, and the definition-wide algorithm would be a good option as well if the inconsistent points can be successfully removed.

Huiqing, Zhang, et al. [7] proposed a process with SURF algorithm and advanced MIC algorithm. First, the image is a smooth variable filter, using the same pixel measurement method to calculate numbers of the same gray value; then adopting a multi-grid algorithm detects geometric angles in non-flat areas of the image; finally, we use the SURF algorithm for matching features.

The authors in [8] proposed MSURF, a new image algorithm that combines Homography with the SURF algorithm. Experiments were performed by comparing MSURF with a standard compatible algorithm based on quantitative and qualitative features, and test results confirm that MSURF can increase not only the value but also the accuracy of matching points.

An efficient approach for video summary without using complex algorithms is proposed in [9]. The system combines the color and edge histogram features with an optimal threshold to detect the keyframes. A genetic algorithm is used as a technique to select the optimum threshold in order to increase the performance of the system.

Thomas, Sinnu Susan, et al. [10] proposed an appropriate summary framework for viewing videos. In addition to reducing the search time, writers propose to turn the problem of finding a content-based video into a content-based retrieval problem. Tests are performed on different data sets to confirm the proposed method of intelligent observation.

Video summarization based on SURF features and its optimization using the Graph theory approach based on objective function are proposed in [11]. The proposed algorithm is tested on two different videos from the Open Video database. The results of the video snapshot and its efficiency found indicate a lack of activity and improvements in the basic understanding of the video snapshot.

Dipti Jadhav, et al. [12] used motion comparison between consecutive frames and then applied geometric relationships to perform video summarization. Movement between frames is represented by affine and homograph changes. Video frames are represented by a set of accelerated dynamic features (SURF). The keyframes are displayed sequentially by comparing the sequence of the previously announced keyframes according to the movement.

The authors in [13] proposed research work that discusses the segmentation and summarization of the frames. A genetic algorithm (GA) for segmentation and summarization is required to view the highlight of an event by selecting a few important frames required. The GA is modified to select only keyframes for summarization and the comparison of modified GA is done with the GA.

3 Proposed System

The proposed system block diagram is shown in Fig. 1. The algorithm steps for the same are stated below.

Steps:

1. Input video for summarization.
2. Extract the frames from the video.
3. Apply SIFT algorithm on the extracted frames to obtain the feature descriptor/vector.
4. Parallel execution of Part A and Part B is performed to obtain keyframes from the feature descriptor.

Part A Steps:

- Calculate the feature difference between consecutive frame feature values of the SIFT descriptor vector obtained in Step 3.
- Store this calculated difference in an array.
- Calculate Mean and Standard Deviation of the array obtained in the previous step using the standard formula.
- Calculate threshold by Niblack's formula given in Eq. (1).
- Compare Threshold and Difference array

if Difference value > TH

Keyframe

else

Ignore and go to next value

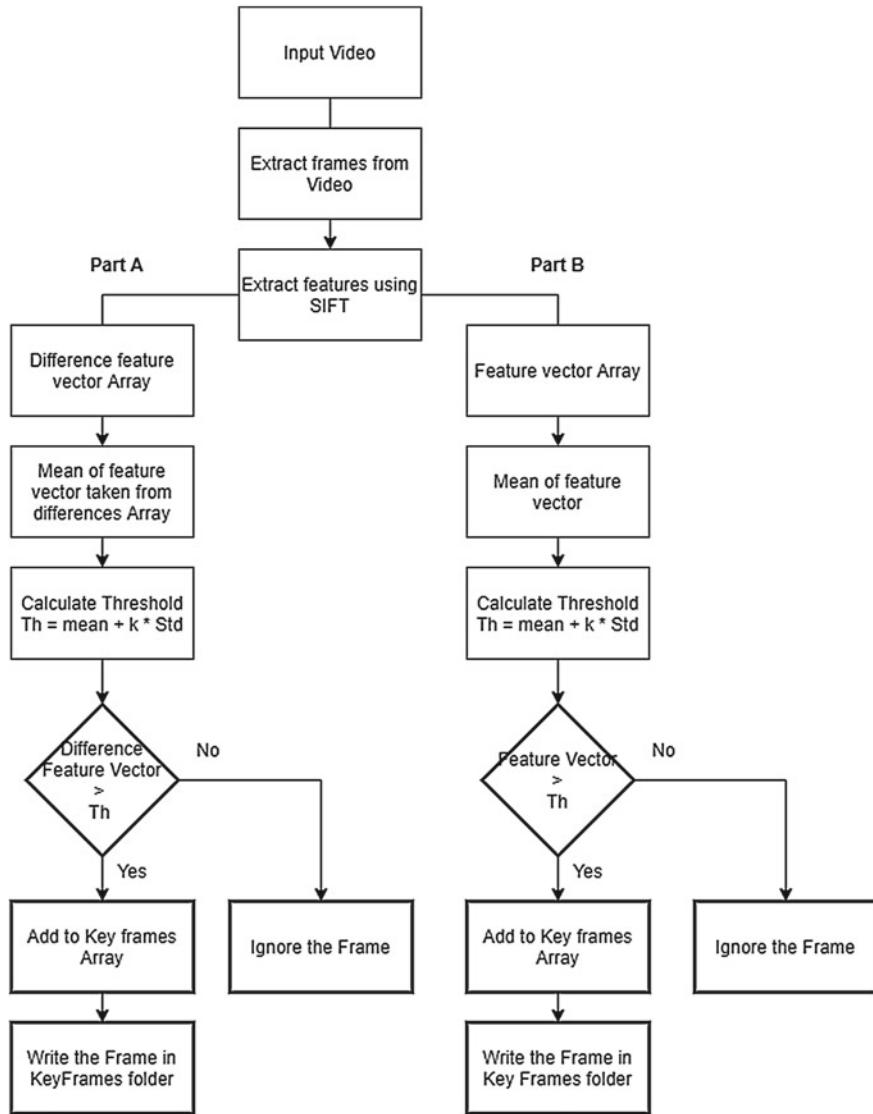


Fig. 1 Block diagram of the proposed system

- Generate the keyframe array by the above method.
- Write the keyframes to a directory.

Part B Steps:

- Calculate Mean and Standard Deviation of the SIFT descriptor obtained in Step 3 using the standard formula.

- Calculate threshold by Niblack's formula given in Eq. (1).
- Compare Threshold and Difference array

if Difference value > TH

Keyframe

else

Ignore and go to next value

- Generate the keyframe array by the above method.
- Write the keyframes to a directory.

The SIFT algorithm is applied on each frame of the video and a feature vector for each frame of the video is obtained. The obtained feature vectors are concatenated into a single file where each row denotes SIFT descriptor features of each frame of the video starting from the first frame to the last frame in the video. This gives a single feature vector representing features of the entire video that is considered.

In the next stage, the obtained vectors undergo a mathematical evaluation to reduce the total frames to a specific set of keyframes as shown in Fig. 1 block diagram.

Part A Flowchart—The method applied for reducing the number of frames and obtaining keyframes is based on feature differencing with thresholding.

The difference of the frames is calculated and stored in a frame difference variable. This obtained difference is used to perform the calculation of mean and standard deviation values. The threshold for the proposed system is obtained by applying Niblack's Thresholding (NBT) formula shown in Eq. (1) below.

$$TH = \text{mean} + k * \text{std} \quad (1)$$

where TH—threshold, std—standard deviation and k—weight that is used to adjust the effect of std.

Part B Flowchart—In the other method, mean and standard deviation are calculated over extracted SIFT features. The threshold value is calculated using the mean and standard deviation, i.e. Niblack's Thresholding (NBT) [14]. Then the feature vectors are compared with this threshold to obtain the keyframes. The K value used as weight is adjusted with various values varying from 0.10 to 0.75, and it is found that K at 0.25 provides effective unique keyframes. The usage of difference in feature vector eliminates the bunch of similar frames closer to one another and just extracts one keyframe out of the 100's. The threshold value comparison helps to create an array of keyframes, and these frames are written in the keyframes folder.

The SIFT algorithm can be divided into four steps:

1. Feature point detection;
2. Feature point localization;
3. Orientation assignment;
4. Feature descriptor;
5. Feature matching.

- The acquisition of key points in the SIFT algorithm is made with the help of the Gaussian Algorithm Variation.
- After obtaining key points, it is important to get local points for accurate results and it is done using the Hessian Matrix (2X2).
- These key local points are assigned to the orientation, a 36i barrel orientation histogram covering 360 degrees.
- The 16X16 area around the key point is taken up and separated by blocks less than 4X4, so a total value of 128 bin is available.
- Finally, the matching of key points is made by the nearest neighbor, and eliminating the level of false games of the closest distance to the nearest distance has been taken.

4 Results and Discussion

Table 1 shows the subjective analysis result which was done with 10 testers showing them the original video and then sharing with them the set of keyframes obtained from Niblack's thresholding with SIFT and the feature differencing applied over Niblack's threshold. These 10 testers selected were students, research colleagues and research students. The obtained values reflect that the proposed system gives satisfactory results on the standard publicly available video data sets.

The subjective analysis was done on 2 custom videos video1_A and Video2_Y using the proposed system. The proposed system is also tested on Open Video database videos v24 and v37 [15], and precision and recall [16] values are calculated. Table 2 shows the values for precision and recall.

The keyframe results obtained from the proposed system were compared with the VSUMM [17] and STIMO [18] results, and the precision and recall table was calculated. The video summarization of the standard open videos V24 and V37 was effectively done using the proposed system.

Table 1 Subjective analysis of videos V1 and V2

Video	Technique	Total frames	Keyframes	Poor	Good	Excellent
Video1_A	SIFT + NBT	384	59	7	1	2
	SIFT + NBT + Diff	384	26		3	7
Video2_Y	SIFT + NBT	378	50	6	4	
	SIFT + NBT + Diff	378	33		2	8

Table 2 Precision and recall for proposed system for videos V24

Videos	Video summarization technique	Precision	Recall
V24	SIFT + NBT + Feature Diff	0.75	1.0
V37	SIFT + NIBT + Feature Diff	0.75	1.0

The redundancy was reduced significantly and unique frames for summarization were obtained.

$$\text{Precision} = \frac{\text{nMatch}}{\text{nVS}} \quad (2)$$

$$\text{Recall} = \frac{\text{nMatch}}{\text{nGT}} \quad (3)$$

where nMatch is the standard framework between global reality and video-based summary, the number of nVs for short video frames and nGT number of frames for ground-based video summary (Figs. 2 and 3).

Comparison of frames between the proposed system and VSUMM shows similarity and matching with good precision and recall values.



(a) Keyframes extracted using proposed technique



(b) Keyframes extracted using VSUMM [17]

Fig. 2 Comparison of proposed system and VSUMM for V24. **a** Keyframes extracted using proposed technique. **b** Keyframes extracted using VSUMM



Fig. 3 Comparison of proposed system and VSUMM for V37

5 Conclusions

In the existing system, there are opportunities to reduce the redundancy of keyframes and increase the accuracy of the summarized video. We propose a solution in order to solve these issues and try to yield better results. In the proposed system, SIFT features are extracted and optimized using the feature difference method. Threshold is calculated with the help of Niblack's threshold and based on this threshold, the difference feature vector comparison is done and keyframes are extracted. The standard videos tested are V24 and V37; the summarized frames were compared with ground truth frames to determine precision and recall. The results are obtained on 2 custom videos with subjective analysis and 2 standard videos with objective analysis. The precision and recall results are very satisfactory and have reduced the redundant frames.

References

1. Z. Elkhattabi, Y. Tabii, A. Benkaddour, Video summarization: techniques and applications. World Academy of Science, Engineering and Technology. Int. J. Comput. Electr. Autom. Control Inf. Eng. **9**, 928–933 (2015)
2. D.G. Lowe, Object recognition from local scale-invariant features, in *1999 Seventh IEEE International Conference on Computer Vision* (1999), pp. 1150–1157. <https://doi.org/10.1109/ICCV.1999.790410>
3. A. Agarwal, D. Samaiya, K.K. Gupta, A comparative study of SIFT and SURF algorithms under different object and background conditions, in *2017 International Conference on Information Technology (ICIT)* (2017). <https://doi.org/10.1109/icit.2017.848>
4. N. Senthilkumaran, C. Kirubakaran, Efficient implementation of Niblack thresholding for MRI brain image segmentation. Int. J. Comput. Sci. Inf. Technol. **5**(2), 2173–2176 (2014)
5. J. Xingteng et al., Image matching method based on improved SURF algorithm, in *2015 IEEE International Conference on Computer and Communications (ICCC)* (IEEE, 2015), pp. 142–45. <https://doi.org/10.1109/CompComm.2015.7387556>

6. Q. Wei et al., Strategies of improving matching accuracy about SURF, in *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)* (IEEE, 2017), pp. 1–6. <https://doi.org/10.1109/EIIS.2017.8298615>
7. Z. Huiqing et al., A fast image matching research based on MIC-SURF algorithm, in *The 27th Chinese Control and Decision Conference (2015 CCDC)* (IEEE, 2015), pp. 542–47. <https://doi.org/10.1109/CCDC.2015.7161751>
8. Z. Saishang et al., MSURF: a new image matching algorithm which combines homography and SURF algorithm, in *2015 23rd International Conference on Geoinformatics* (IEEE, 2015), pp. 1–7. <https://doi.org/10.1109/GEOINFORMATICS.2015.7378709>
9. A. Phadikar et al., Video summarisation using optimum global threshold technique based on genetic algorithm. *Int. J. Innov. Comput. Appl.* **7**(1), 1 (2016). <https://doi.org/10.1504/IJICA.2016.075457>
10. S.S. Thomas et al., Smart surveillance based on video summarization, in *2017 IEEE Region 10 Symposium (TENSYMP)* (IEEE, 2017), pp. 1–5. <https://doi.org/10.1109/TENCONSpring.2017.8070003>
11. D. Jadhav, U. Bhosle, SURF based video summarization and its optimization, in *2017 International Conference on Communication and Signal Processing (ICCP)* (IEEE, 2017), pp. 1252–57. <https://doi.org/10.1109/ICCP.2017.8286581>
12. D. Jadhav, U. Bhosle, Video summarisation based on motion estimation using speeded up robust features. *Int. J. Comput. Vis. Robot.* **9**(6), 569 (2019). <https://doi.org/10.1504/ijcvr.2019.104039>
13. H.S. Prashantha, Video segmentation & summarization using modified genetic algorithm. *Int. J. Comput. Sci. Appl.* **8**(4/5), 01–09 (2018). <https://doi.org/10.5121/ijcsa.2018.8501>
14. S. Farid, F. Ahmed, Application of Niblack's method on images, in *2009 International Conference on Emerging Technologies* (Islamabad, 2009), pp. 280–286. <https://doi.org/10.1109/ICET.2009.5353159>
15. Sites.google.com (2020). Download—VSUMM (Video Summarization). <https://www.sites.google.com/site/vsummsite/download>. Accessed 21 Nov 2020
16. S. Mei, G. Guan, Z. Wang, S. Wan, M. He, D.D. Feng, Video summarization via minimum sparse reconstruction. *Pattern Recognit.* (2014). <https://doi.org/10.1016/j.patcog.2014.08.002>
17. S.E.F. de Avila, A. da Luz Jr., A.A. Araújo, M. Cord, VSUMM: an approach for automatic video summarization and quantitative evaluation, in *Proceedings of the 2008 XXIBrazilian Symposium on Computer Graphics and Image Processing* (12–15 October 2008), pp. 103–110. <https://doi.org/10.1109/SIBGRAPI.2008.31>
18. M. Furini, F. Geraci, M. Montangero, M. Pellegrini, (2010) STIMO: still and moving video storyboard for the web scenario. *Multimed. Tools Appl.* **46**(1), 47–69. <https://doi.org/10.1007/s11042-009-0307-7>

A Comprehensive Study on Attention-Based NER



Tanvir Islam, Sakila Mahbin Zinat, Shamima Sukhi, and M. F. Mridha

Abstract Named Entity Recognition (NER) is a part of extraction and is used for Natural Language Processing (NLP). NER system helps us to find various names from unstructured text or a text file and classifies them into various categories. The attention-based keyword extracting concept has been established to solve the problem of detecting redundant data and inessential data and does not consider them. Researchers are highly concerned about attention mechanisms. In this study, we focus on the most recent algorithms which are trained with the attention-based mechanism for NER. We briefly describe attention-based models, objectives of these models, datasets used in each method, and efficiency. Our focus is to give some decisions on which model is exceptionally efficient depending on the dataset and NER category.

Keywords Named Entity Recognition · Natural Language Processing · Attention · Mechanism · Deep learning

1 Introduction

Human beings can easily find out the main concepts or main keywords that should be captured for a specific purpose. The brain of humans can capture any circumstances based on whatever the needs should be. But when any researcher wants to do all of these works by any machine, then he has to train it by implementing an algorithm or procedure through which it will work and give the predicted result. NER works for various reasons but its basic concern is to extract all the names from a text file or unstructured data and classify them into various categories. The categories can be person, location, organization, company, product, and so on. In recent previous years, we have seen NER for medical purposes like medicines' name extraction, diseases' name extraction, medical equipment's name extraction, and medical-tests' name extraction. This kind of NER is so much helpful in the real world. NER can

T. Islam · S. M. Zinat · S. Sukhi · M. F. Mridha (✉)

Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka, Bangladesh
e-mail: firoz@bubt.edu.bd

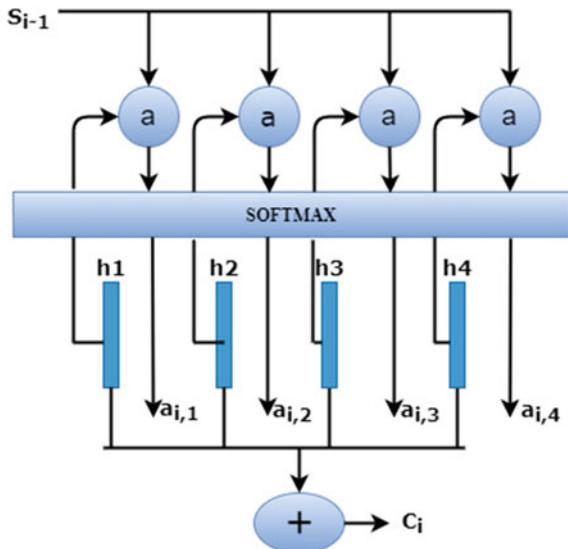
help to summarize the context and also helpful for understanding the context. NER is a small part of NLP. NLP has some components to process such as syntax, semantics, and speech. NER works in the semantics of NLP. The semantics part of NLP is for understanding the meaning of sentence word and their relationship. In our study, we have seen many kinds of the mechanism of NER such as Ontology-based NER, Deep Learning-based NER, Long Short-Term Memory (LSTM)-based NER, Bidirectional-LSTM (Bi-LSTM) NER, BiLSTM-Conditional Random Field (CRF) NER, Deep Neural Network-based NER, and so on. We focus on Attention-based NER. Many researchers include Attention-based mechanisms with their proposed model like Attention-based BiLSTM-CRF NER.

The attention mechanism was first applied by Bahdanau et al. for translating aligning on machine translation works simultaneously [1]. Researchers use an Attention-based mechanism because it has a large amount of memory. Many models can't store the vast amount of input sequences. The attention mechanism maintains the sequence of input and output and builds cooperation between them. This model has a hidden layer where the input sequence is stored, and the context or output term has the chance to look backward for correlation with the input sequence.

So, here is the visualization of the attention mechanism. In Fig. 1, this is the soft alignment of the attention mechanism. Here, input context stores a probability value for each element in the hidden layer (h) that we call memory and alpha is the weight or distribution function. It will deliver output depending on the probability and distribution. Many models or algorithms use the concept of attention based on NER. Especially, we have seen many of them use Bi-LSTM and BiLSTM-CRF.

There exist a lot of works with NER. Attention-based NER models perform idly based on different purposes. But it is difficult for a researcher who wants to establish

Fig. 1 Basic architecture of attention-based mechanism (soft alignment)



a model for attention-based NER because he has to review several research works for making the decision, and it will be somewhat time-consuming and he may not be able to visualize everything parallelly as to which model will be suitable for his work. These circumstances make us motivated to study NER for extracting keywords and notice that attention-based NER with deep learning models gives more efficiency than other NER.

Focusing on such complication, we decided to provide this study where only attention-based NER models and some of the recent most efficient NER approaches have been mentioned all together till 2020 and as a result, the researcher may effortlessly make a decision based on his needs. The exact contribution of our study can be condensed as follows:

- We illustrate a better visualization of attention-based NER models' efficiency and their objects.
- We are the first conductor who has mentioned the recent NER works which are based on attention-based models only.
- We try to make understand some best model outcomes with the attention-based mechanism by reviewing only this study.

It is complicated for us to decide which deep learning models are best for all extracting keyword tasks and detecting named entities, and we cannot make a decision properly which one gives the most superior efficiency though efficiency varies from different datasets. In the future, we will study more deeply to analyze and decide which one is the most proficient among all of the attention-based deep learning models.

This study is partitioned in this way. In this study, Sect. 2 illustrates the previous works of extracting NER based on the attention-based mechanism. Table 1 shows a visualized scenery where we mentioned the efficiency and objectives of some most recent applied or implemented attention-based models which will help researchers to make a decision as to which model will be suitable for their work. The role of the attention mechanism has been illustrated in Sect. 3. We have reviewed some ancient research works also which were established through Biomedical, Clinical, and Electronic Medical Records (EMR), and give an idea of their performance level through some graphs in Sect. 4. Section 5 condenses the most recent attention-based models which will help the researchers for deciding on their works which should be about attention-based NER. The overview of this study has been illustrated in Fig. 2.

2 Attention-Based Models

2.1 LSTM-CRF

Long Short-Term Memory (LSTM) is a particular type of Recurrent Neural Network (RNN) that can learn long-term dependencies. It remembers the long durations of

Table 1 Overview of the attention-based models containing objectives and advantages of the models

Models	Objectives	Advantages
LSTM-CRF	(1) Classifying aspect-level embeddings and sentiment classification [1] (2) Attaining global information from given sentences of interest [2] (3) Extracting both coherent and incoherent entities especially for Clinical Named Entity Recognition in the Chinese language [3]	(1) It can find the exact meaning of different types of aspects based on different situations [1] (2) Can be used for natural language processing for various languages [2] (3) Can determine relativity and adopt local context information [3]
BiLSTM-CRF	(1) Linking entities [1] (2) Generating embeddings of each mention and its candidate entities [1] (3) Labeling the final sequence. Solving the bias problem in labeling [4] (4) Obtaining the global optimal label for CRF [4] (5) Detecting document-level correlation [5] (6) Effectiveness in both word-based and character-based [7]	(1) Can extract semantic correlative features of mention and entity [1] (2) Can extract bidirectional features [1] (3) Can extract long-distance dependencies [2] (4) Word composition, word-element segmentation is beneficial [3] (5) Can reduce the inconsistency of tagging [3] (6) Can solve sequential modeling [5]. Work very well in Chinese NER (7) Can prevent information loss for attention mechanism [8] (8) Can give better performance in the intent analysis compared with other methods [9] (9) Can give high performance on knowledge-based NER [9]
BiLSTM-CCAtt	To define the familiarity between two exact entities from a sentence	(1) Can extract keywords from large-scale datasets especially for the Chinese language (2) Can produce very efficient information about Chinese relation extraction task

(continued)

Table 1 (continued)

Models	Objectives	Advantages
BiGRU	<ul style="list-style-type: none"> (1) To extract NER for geological hazard literature [1] (2) To work with pattern-based corpus [1] (3) To use for Intent classification [2] (4) To help for Slot filling tasks [2] (5) To use for Question classification task [3] 	<ul style="list-style-type: none"> (1) Learns automatically and transforms features [1] (2) Different depth of BiGRU layer can extract different features [1] (3) Can also enhance the features for using an attention-based mechanism [1] (4) Can learn the context information [3] (5) Can read information both in forward and reverse directions [3] (6) Can capture sequence context feature [4] (7) Attention mechanism gives more focus on the hidden layer of BiGRU [4] (8) DNN-BiGRU works much better than BiLSTM neural network based on the character relation mining method [5]
AERNs	To find nested named entities as well as non-overlapped NER successfully	<ul style="list-style-type: none"> (1) Can perform better to Recognize Nested Named Entity Recognition (Nested NER) rather than other models (2) Can experiment on possible entity regions and classify them

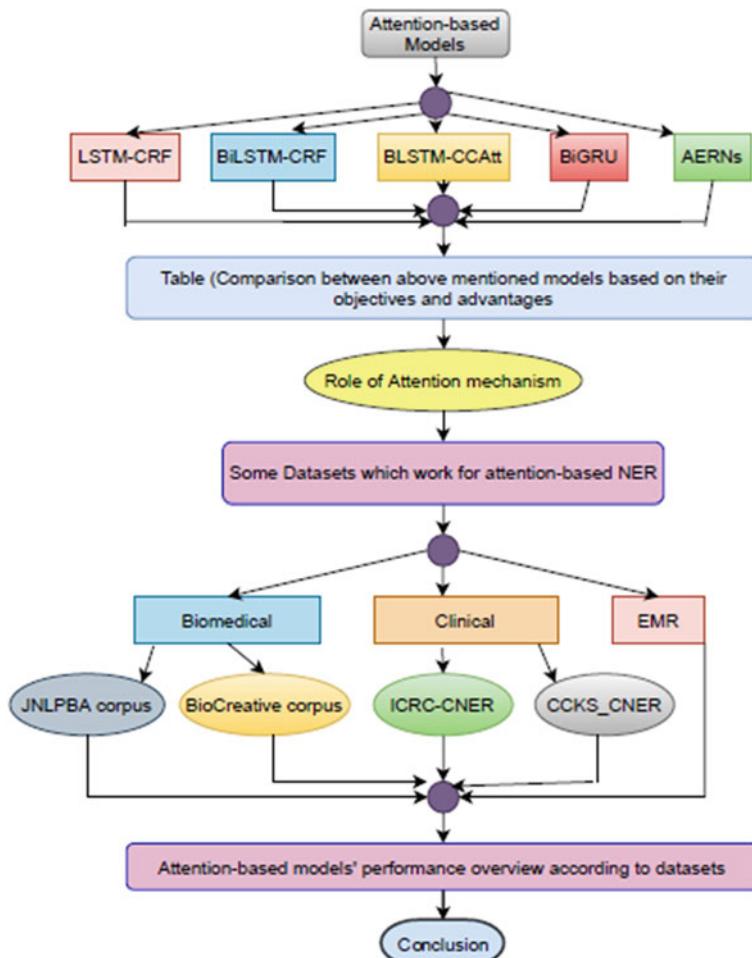


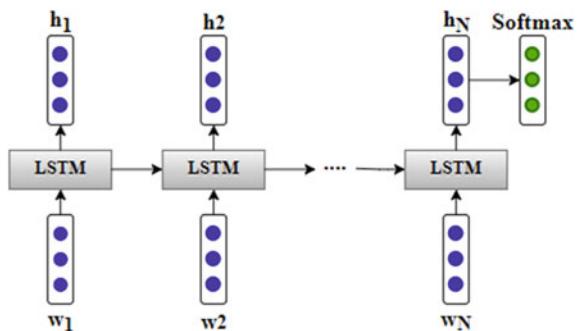
Fig. 2 Overview of this study

time. The architecture of standard LSTM illustrates that it has a cell memory state and three gates which are shown in Fig. 3.

Here, $\{h_1, h_2, \dots, h_N\}$ are the hidden vectors and $\{w_1, w_2, \dots, w_N\}$ are the word vectors of a sentence with length N which can be considered as weighted matrices. The last hidden vector h_N can be considered as a representation of a sentence that places into the *softmax* layer after being linearized into a vector. The value of the vector's length should be similar to the class labels' number where the set of the class labels can be {positive, positive, negative}, {positive, negative, neutral}, {negative, positive, neutral}, etc.

The problem of detecting essential parts of aspect-level sentiment classification by the primary LSTM model can be solved by the attention mechanism that can detect

Fig. 3 The architecture of a standard LSTM



the key part of a sentence to a mentioned aspect. This type of work was proposed by Yequan Wang et al. about Attention-based Long Short-Term Memory (LSTM) Network for this aspect-level sentiment classification which can provide complete and appropriate results [2].

Conditional Random Field (CRF) is a kind of preferential model which is suitable to predict tasks where applicable information of neighbours sentiment the current prediction. CRF applies to Named Entity Recognition (NER). A combination of the LSTM network and CRF network can composite the LSTM-CRF model which can be efficiently used in previous input features by the LSTM layer and tagging information by the CRF layer. Zengjian Liu et al. proposed a research work which was developed by the attention-based Convolutional Neural Network-LSTM-CRF model where an entity recognition for Chinese clinic is in a neural network [3]. This model was established for making global information of each word and sentence.

Another work was proposed with the Attention-based CNN-LSTM-CRF model for recognizing both adjoining and disjoining clinical texts in the Chinese language [4].

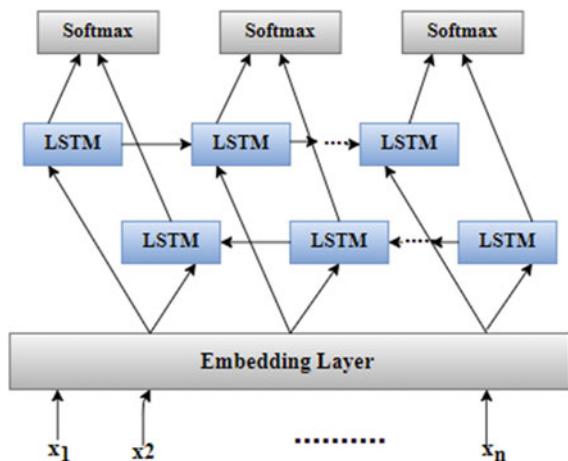
2.2 BiLSTM-CRF

Bidirectional LSTM or Bi-LSTM is a circumambulation processing model which comprises two LSTMs: one is taking input in a forward direction, which means passing input from past to future, and the other one in a backward manner, which means from future to past. The bidirectional LSTM works bidirectionally for preserving information in both forward and backward processes which are shown in Fig. 4.

Shengze et al. proposed SA-ESF, which buttresses the symmetrical Bi-LSTM neural network with a dual attention mechanism for calculating the interrelation between mentions and entities in two perspectives [5].

State-of-the-art (or close to) accuracy, chunking, and NER datasets can be produced by the BiLSTM-CRF model. It is sturdy and has a lower dependence

Fig. 4 Bidirectional LSTM network



on word embedding than previous works. The combination of the Bi-LSTM and CRF models gives the best output for finalizing the sequence labeling. This model is used broadly in many research works. We pick one of the best works in those to use the BiLSTM-CRF model.

Zhiwei Yang et al. proposed a research work about Attention-based Multi-level Feature Fusion (AMFF) [6]. Their study is about conducting it for the purpose of capturing the multi-level features to improve due to various perspectives and designed four components to capture the features from the perspectives of capitalization, inter-word relations, keywords, and lexical phrases, respectively.

Mingyan Tan et al. proposed a research work which is “An Attention-Based Approach for Mongolian News Named Entity Recognition” [7]. Their work was according to the characteristics of a Mongolian, a Named Entity Recognition (NER) which is based on attention mechanism, where the model is a traditional model that is formed to use the Conditional Random Field (CRF) and Bidirectional Long-Short Term Model (LSTM) methods.

Identification of entities and relationship on Chinese EMR was proposed by Menglong Li et al. which was implemented by the BiLSTM model [8]. The EMR labeling bias problem was solved by their work where they employed the transformation matrix in CRF to solve it.

Attention-based Bidirectional Long-Short Term with Conditional Random Field (Att-BiLSTM-CRF) was used in another research work where Named Entity Recognition (NER) to extracting entities was applied for describing from geoscience reports to geoscience information [9].

Another work was researched for sentence-level attention mechanism for a Named Entity Recognition (NER) task in the Chinese language [10]. This work proved that adding an attention mechanism improves the impact and efficiency of the NER, and constraints between labels are increased by the CRF layer.

Word and character-based information can be used in a better way by the BiLSTM-CRF model which should be based on an attention-based mechanism. Chaoyi Huang et al. proposed a research work about word-level and character-level embeddings which helps to make proper use of word information and producing a weight vector in the hidden layer (attention layer) [11].

The biomedical text mining with NER is one of the bottom tasks of extracting information. The attention-based BiLSTM-CRF model is being rapidly used for Biomedical Named Entity Recognition (BNER). Hao Wei et al. established this model for BNER for the improvement to identify biomedical entities where the loss of important information was prevented by this model [12].

Another work has been done for building an online medical question answering system in the Chinese language by Chaochen Wu et al. [13]. They used this model for NER for integrating text classification and sequence tagging tasks.

A chemical entity is an essential part of entities in Biomedical research. Several neural network approaches have been proposed with the Att-BiLSTM-CRF model. Ling Luo et al. proposed a research work that is based on a document-level attention mechanism where it allows the detection of the related tokens of different sentences as a tagging problem which is dependent [14].

Luqi Li et al. proposed a research work for eliminating semantic interference and improving the ability of autonomous learning of internal features [15]. They integrated BiLSTM-Att-CRF which is an improved method of named entity recognition for Chinese electronic medicine. More useful information can be captured by these records. The validity of the attention mechanism is discovering key information, and mining text features is confirmed by their research work.

Learning text representation, achieving state-of-the-art performance, and outperforming previous works made Vaswani et al. to use the self-attention mechanism [16]. The self-attention mechanism was used in another work proposed by Guohua Wu et al. for Clinic Named Entity Recognition (CNER) task in the Chinese language [17]. The challenge of capturing long-range dependencies was addressed by them where they proposed the Att-BiLSTM-CRF model using a fine-grained character-level method.

Relation extraction on capsule network for learning framework (multi-label) was explored by Ningyu Zhang et al. with an attention mechanism where they worked for building the process of delivering messages from root nodes to targeted nodes [18].

Another invention has been done by Tao Li et al. where they built a self-attention-based BiLSTM-CRF model for extracting entities which is related to cybersecurity where it converts structured data to textual information [19].

Lee, J. et al. proposed an effective research work that improves entities and their dormant types as features and constructs word representations which is based on the self-attention symmetrical similarity of a sentence itself [20].

2.3 BLSTM-CCAtt

Xiaoyu Han et al. proposed a research work that identifies the relationship between two specified entities in a sentence [21]. They generated a large-scale Chinese relation extraction using the attention-based model. They proposed a neural network model which is named as Bidirectional-Long Short Term Model using Character Composition Attention (BLSTM-CCAtt). They found that model more effective using an attention-based model which can find out a vital part of a sentence.

2.4 BiGRU

Bidirectional Gated Recurrent Unit (BiGRU) is formed with two GRUs: one takes input (reading text, extracting word context information, etc.) in the forward direction and another one is in the backward direction which is connected with an equivalent output layer as shown in Fig. 5.

GRU is formed of an update and reset gate where the network parameters are made low and converged more easily by it.

Fan, R. et al. proposed a research work that extracts geological hazard named entities from the important body of geological hazard literature [22]. They built a geological hazard knowledge graph based on the NER model, namely the deep, multi-branch BiGRU-CRF model that makes a combination between multi-branch Bidirectional Gated Recurrent Unit (BiGRU) layer and a Conditional Random Field (CRF) model.

A formatted medical record receives the conversation between a doctor and patient and delivers it to the appropriate department for extracting entities, which was done by Yuming Li et al. where they build a BiGRU-CRF model [23].

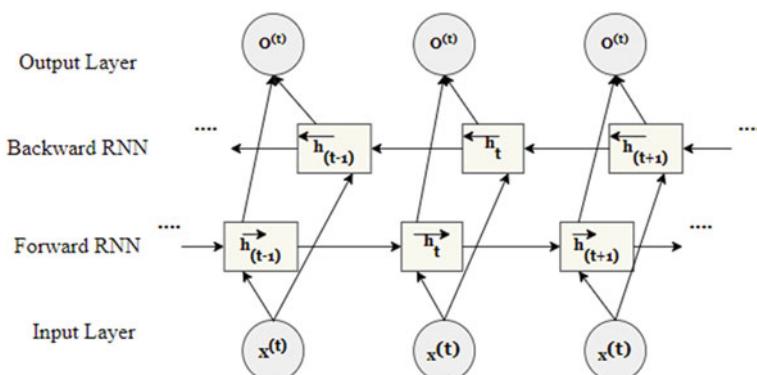


Fig. 5 Structure of BiGRU

The BiGRU model is also being used for the question classification task. One of this type of work has been done by Jin Liu et al. where they build a BiGRU-CNN network based on attention mechanism in the Chinese language [24].

Another work has been done by Jianfeng Deng et al. where they proposed a BiGRU network for capturing context information and removed the problem of redundancy through a self-attention mechanism [25].

A joint model such as the DNN-BiGRU network helps to combine part-of-speech and information of location for building a knowledge graph. This type of work has been done by Ying He et al. where they build a joint model which is based on the DNN-BiGRU network for recognizing names of persons and extracting character relation which is important for obviousness poverty palliation [26].

2.5 AERNs

Dai et al. proposed a model Attention-Based Entity Region Networks (AERNs) about multi-grained entity recognition where the coarse-grained named entity overlaps with the fine-grained named entity [27]. AERNs is a neural framework. Attention-Based Entity Region Networks (AERNs) is a multi-grained entity recognition where a sentence could not overlap or create nested. In AERNs, entity region classifiers classify the regions. Using the attention mechanism, context information is included for enhancing the execution of NER and it helps to identify the context information. This model is also natural and intuitionial which is extract-related information and it can be potentially beneficial. AERNs build two sub-modules, one is the Entity Region Recognizer and the second one is Entity Region Classifier. The deep neural network is the Recognizer. Within the input sentence region, the recognizer recognizes all the potential entity regions with nested and non-overlapping entities. And the Entity Region Classifier classifies the entity regions into pre-defined classes. At last, an attention mechanism captured context information which helps to utilize entity-related context information.

3 Role of Attention Mechanism in NER

Attention-based models are so efficient models that give better performance in NER. Table 1 illustrates some common model's objectives and advantages that are used in NER. From Table 1, any researcher can easily understand the role of the attention mechanism in NER and which models have been used and perform better. Every model of text extraction performs much better for using the attention mechanism. Attention mechanism can help to extract more features in NLP. The BiLSTM-CRF model performs much better in NER according to our analysis and when attention layer works also with Bilstm-CRF model, it can capture more depth features. NER platform is now a vast research area for question answering but NER with attention

mechanism is a very well-performed model. So, this paper shows a detailed study of attention-based models in NER.

4 Most Used Datasets in NER

Here, we tried to give a visualized idea by giving graphs that show which models are frequently used in attention-based NER. The graphs consist of the percentages according to F1-score accuracy measurement of the used models for NER. Our analysis will help to understand which model has been used the most and performs more efficiently according to dataset type. We tried to represent the role of the most used datasets especially for the medical sector and also based on the attention-based mechanism.

JNLPBA Corpus: JNLPBA is an open-source corpus that has been annotated in the biomedical keyword extraction task. It comprises five entity types such as Cell Type, RNA, DNA, Protein, and Cell Line (Table 2).

BioCreative Corpus: BioCreative corpus (<http://www.biocreative.org/resources/>) is organized by the collaboration between information extraction systems and text mining, and it is embedded in the biological state (Table 3).

ICRC_CNER Dataset: 1176 medical records are physically annotated on five groups of clinical substances in the ICRC_CNER dataset. It also includes 176 records as an advancement set, 600 of which are utilized as a training set and others are as a test set including continual and dissociated clinical entities (Table 4).

Table 2 List of some research works based on attention-based models on JNLPBA corpus with reference

Models	Authors
Attention-based BiLSTM-CRF	Zhiwei Wang et al. [6]
	Hao Wei et al. [12]
	Marek Rei et al. [28]
	Usman Naseem et al. [29]
	Harsh Patel [30]
	Hao Wei et al. [31]

Table 3 List of some research works based on attention-based models on BioCreative corpus with reference

Models	Authors
Attention-based BiLSTM-CRF	Ling Luo et al. [14]
	Luqi Li et al. [15]
	Yung Pei et al. [32]
	Wahed Himati et al. [33]
	Zhijing Li et al. [34]

Table 4 List of some research works based on attention-based models on the ICRC_CNER dataset with reference

Models	Authors
CNN-LSTM-CRF	Zengjian Liu et al. [3]
	Buzhou Tang et al. [4]

Table 5 List of some research works based on attention-based models on CCKS_CNER datasets with reference

Models	Authors
CNN-LSTM-CRF	Buzhou Tang et al. [4]
BiLSTM-Att-CRF	Luqi Li et al. [15]
	Guohua Wu et al. [17]
	Xianglong Chen et al. [35]
ELMo-ET-CRF	Qian Wan et al. [36]

Table 6 List of some research works based on attention-based models on EMR data with reference

Models	Authors
BiLSTM-Attended-CRF	Menglong Li et al. [8]
	Lejun Gong et al. [37]
SM-LSTM-CRF	Xiaoling Cai et al. [38]

CCKS_CNER Datasets: It is a clinical entity that has features like extraction of disease, symptoms, tests, treatments, and medication. It has been annotated exactly for contiguous and discontiguous clinical entities (Table 5).

EMR data: Electronic Medical Records are formed of digital identical patient data which helps organizations to serve with proficient and proper care (Table 6).

5 Discussion

By visualizing all recently used attention-based models, we have found that the Bi-LSTM model has the double attention mechanism entity linking system that gives the benefit of entity embeddings and mentions context, embedding, and entity description. If we concentrate on entity context information, another model is AERNs. In the AERNs model, a sentence couldn't include or create nested or non-overlapping entities but if we use an attention-based entity region network and it is multi-grained entity recognition and this model also captured context information to exploit entity related context information. In the BLSTM-CCAtt model, we have seen that Xiaoyu Han et al. proposed a model that is best for comparing with other relation extraction tasks, and it can elaborate the relation [21]. Bi-LSTM-CRF model with attention mechanism gives better performance especially for biomedical, clinical, and EMR datasets according to our analysis which has been mentioned already in Figs. 6, 7,

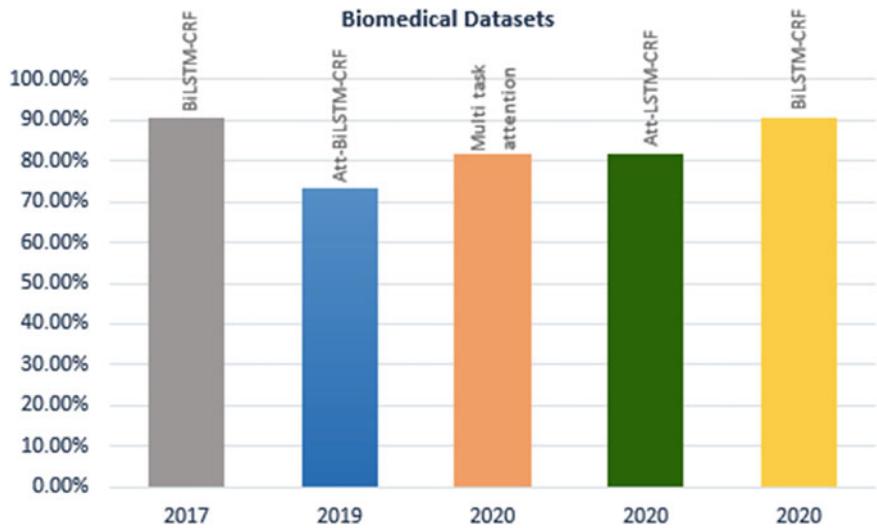


Fig. 6 The accuracy level of attention-based models with Biomedical datasets of some recent years

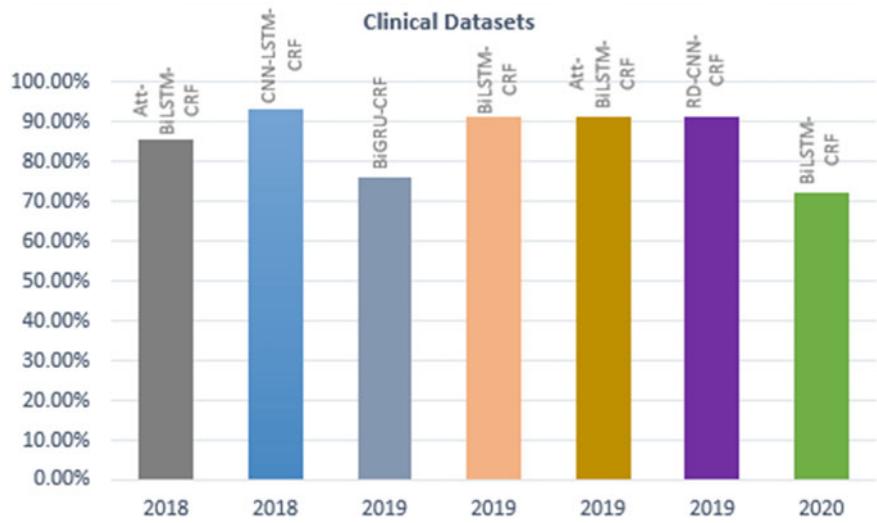


Fig. 7 The accuracy level of attention-based models with Clinical datasets of some recent years

and 8. But the accuracy of these models depends on which purpose the models will be trained and also varies depending on datasets. Here, we pick some best models on attention-based NER altogether in one study. We can't say that any particular proposed model is the best.

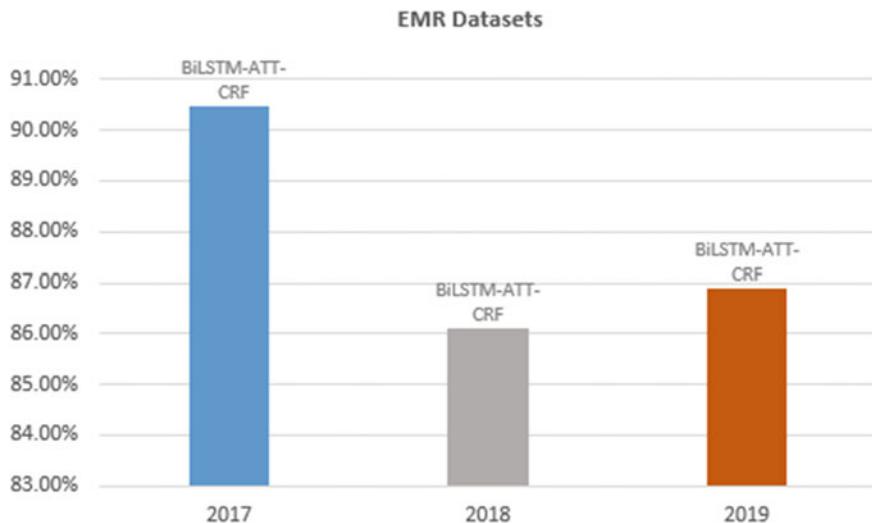


Fig. 8 The accuracy level of attention-based models with EMR of some recent years

6 Conclusion

The attention-based mechanism is used in NER for the increasing efficiency of NER. This mechanism has been conducted in many models for research work that is based on NER. We have picked some of the best models for our study. The researchers will get a brief idea about NER by using the most recent attention-based deep learning models, the previous works through these methods, datasets, and the accuracy rate of each model's performance. Our study will help any researcher for utilizing whichever model will give better efficiency in attention-based NER mechanisms according to the researchers' motive.

References

1. D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate (2014). [arXiv:1409.0473](https://arxiv.org/abs/1409.0473)
2. Y. Wang et al., Attention-based LSTM for aspect-level sentiment classification, in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (2016). <https://doi.org/10.18653/v1/d16-1058>
3. Z. Liu et al., Chinese clinical entity recognition via attention-based CNN-LSTM-CRF, in *2018 IEEE International Conference on Healthcare Informatics Workshop (ICHI-W)* (IEEE, 2018). DOI: <https://doi.org/10.1109/ichi-w.2018.00023>
4. B. Tang et al., Entity recognition in Chinese clinical text using attention-based CNN-LSTM-CRF. *BMC Med. Inform. Decis. Mak.* **19**(3), 74 (2019). <https://doi.org/10.1186/s12911-019-0787-y>

5. S. Hu et al., Entity linking via symmetrical attention-based neural network and entity structural features. *Symmetry* **11**(4), 453 (2019). <https://doi.org/10.3390/sym11040453>
6. Z. Yang et al., Attention-based Multi-level Feature Fusion for Named Entity Recognition. <https://doi.org/10.24963/ijcai.2020/497>
7. M. Tan et al., An attention-based approach for Mongolian news named entity recognition, in *China national conference on Chinese computational linguistics* (Springer, Cham, 2019). https://doi.org/10.1007/978-3-030-32381-3_35
8. M. Li et al., named entity recognition in Chinese electronic medical record using attention mechanism, in *2019 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, 2019). <https://doi.org/10.1109/ithingsgreencomcpcsmartdata.2019.00125>
9. Q. Qiu et al., BiLSTM-CRF for geological named entity recognition from the geoscience literature. *Earth Sci. Inform.* **12**(4), 565–579 (2019). <https://doi.org/10.1007/s12145-019-00390-3>
10. Z. Wan et al., BiLSTM-CRF Chinese named entity recognition model with attention mechanism. *J. Phys.: Conf. Ser.* **1302**(3) (IOP Publishing, 2019). <https://doi.org/10.1109/icccbda49378.2020.9095727>
11. C. Huang, Y. Chen, Q. Liang, Attention-based bidirectional long short-term memory networks for Chinese named entity recognition, in *Proceedings of the 2019 4th International Conference on Machine Learning Technologies* (2019). <https://doi.org/10.1145/3340997.3341002>
12. H. Wei et al., Named entity recognition from biomedical texts using a fusion attention-based BiLSTM-CRF. *IEEE Access* **7**, 73627–73636 (2019). <https://doi.org/10.1109/access.2019.2920734>
13. C. Wu et al., An attention-based multi-task model for named entity recognition and intent analysis of Chinese online medical questions. *J. Biomed. Inform.* **108**, 103511 (2020). <https://doi.org/10.1016/j.jbi.2020.103511>
14. L. Luo et al., An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition. *Bioinformatics* **34**(8), 1381–1388 (2018). <https://doi.org/10.1093/bioinformatics/btx761>
15. L. Li et al., An attention-based deep learning model for clinical named entity recognition of Chinese electronic medical records. *BMC Med. Inform. Decis. Mak.* **19**(5), 235 (2019). <https://doi.org/10.1186/s12911-019-0933-6>
16. A. Vaswani et al., Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30**, 5998–6008 (2017)
17. G. Wu et al., An attention-based BiLSTM-CRF model for Chinese clinic named entity recognition. *IEEE Access* **7**, 113942–113949 (2019). <https://doi.org/10.1109/access.2019.2935223>
18. N. Zhang et al., Attention-based capsule networks with dynamic routing for relation extraction (2018). [arXiv:1812.11321](https://arxiv.org/abs/1812.11321). <https://doi.org/10.18653/v1/d18-1120>
19. T. Li, Y. Guo, A. Ju, A self-attention-based approach for named entity recognition in cybersecurity, in *2019 15th International Conference on Computational Intelligence and Security (CIS)* (IEEE, 2019). <https://doi.org/10.1109/cis.2019.00039>
20. J. Lee, S. Seo, Y.S. Choi, Semantic relation classification via bidirectional LSTM networks with entity-aware attention using latent entity typing. *Symmetry* **11**(6), 785 (2019). <https://doi.org/10.3390/sym11060785>
21. X. Han et al., An attention-based model using character composition of entities in Chinese relation extraction. *Information* **11**(2), 79 (2020). <https://doi.org/10.3390/info11020079>
22. R. Fan et al., Deep learning-based named entity recognition and knowledge graph construction for geological hazards. *ISPRS Int. J. Geo-Inf.* **9**(1), 15 (2020). <https://doi.org/10.3390/ijgi9010015>
23. Y. Li et al., A joint model of clinical domain classification and slot filling based on RCNN and BiGRU-CRF, in *2019 IEEE International Conference on Big Data (Big Data)* (IEEE, 2019). <https://doi.org/10.1109/bigdata47090.2019.9005449>

24. J. Liu et al., Attention-based BiGRU-CNN for Chinese question classification. *J. Ambient Intell. Humaniz. Comput.* 1–12 (2019). <https://doi.org/10.1007/s12652-019-01344-9>
25. J. Deng, L. Cheng, Z. Wang, Self-attention-based BiGRU and capsule network for named entity recognition (2020). [arXiv:2002.00735](https://arxiv.org/abs/2002.00735)
26. Y. He, H. Yun, L. Lin, The character relationship mining based on knowledge graph and deep learning, in *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)* (IEEE, 2019). <https://doi.org/10.1109/bigcom.2019.00011>
27. J. Dai et al., AERNs: attention-based entity region networks for multi-grained named entity recognition, in *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)* (IEEE, 2019). <https://doi.org/10.1109/ictai.2019.00064>
28. M. Rei, G.K.O. Crichton, S. Pyysalo, Attending to characters in neural sequence labeling models (2016). [arXiv:1611.04361](https://arxiv.org/abs/1611.04361)
29. U. Naseem et al., Biomedical named-entity recognition by hierarchically fusing BioBERT representations and deep contextual-level word-embedding, in *2020 International Joint Conference on Neural Networks (IJCNN)* (IEEE, 2020). <https://doi.org/10.1109/ijcnn48605.2020.9206808>
30. H. Patel, BioNerFlair: biomedical named entity recognition using flair embedding and sequence tagger (2020). [arXiv:2011.01504](https://arxiv.org/abs/2011.01504)
31. H. Wei et al., Biomedical named entity recognition via a hybrid neural network model, in *2019 IEEE 14th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)* (IEEE, 2019). <https://doi.org/10.1109/iske47853.2019.9170320>
32. Y. Pei et al., An attention-based approach for chemical compound and drug named entity recognition. *J. Comput. Res. Dev.* **55**(7), 1548 (2018). <https://doi.org/10.7544/issn1000-1239.2018.20170506>
33. W. Hemati, A. Mehler, LSTMVoter: chemical named entity recognition using a conglomerate of sequence labeling tools. *J. Cheminform.* **11**(1), 3 (2019). <https://doi.org/10.1186/s13321-018-0327-2>
34. Z. Li et al., Bio-semantic relation extraction with attention-based external knowledge reinforcement. *BMC Bioinform.* **21**, 1–18 (2020). <https://doi.org/10.1186/s12859-020-3540-8>
35. X. Chen et al., Improving the named entity recognition of Chinese electronic medical records by combining domain dictionary and rules. *Int. J. Environ. Res. Public Health* **17**(8), 2687 (2020). <https://doi.org/10.3390/ijerph17082687>
36. Q. Wan et al., A self-attention based neural architecture for Chinese medical named entity recognition. *Math. Biosci. Eng.* **17**(4), 3498 (2020). <https://doi.org/10.3934/mbe.2020197>
37. L. Gong, Z. Zhang, S. Chen, Clinical named entity recognition from Chinese electronic medical records based on deep learning pretraining. *J. Healthc. Eng.* **2020** (2020). <https://doi.org/10.1155/2020/8829219>
38. X. Cai, S. Dong, H. Jinlong, A deep learning model incorporating part of speech and self-matching attention for named entity recognition of Chinese electronic medical records. *BMC Med. Inform. Decis. Mak.* **19**(2), 101–109 (2019). <https://doi.org/10.1186/s12911-019-0762-7>

Sentiment Analysis of Multilingual Mixed-Code, Twitter Data Using Machine Learning Approach



Sowmya Swamy, Jyoti Kundale, and Dipti Jadhav

Abstract Sentiment Analysis is a process of computing, identifying, and categorizing opinion from a piece of text and discovering if the writer's attitude toward a particular topic or product is positive or negative. In this paper, we are doing Sentiment Analysis of Twitter Data which are mixed code (English and Hindi). Our model works on different types of Twitter input data, namely Mixed script—the tweet is written in combination of Native script and Roman script, Code-Mixed script—in this type, the tweet is in native language but Roman script was used to write it, and Native script—here, the Regional language tweet is written in Native script. We are developing our model in Jupyter Notebook which is an open-source web application software. Our main aim is to increase the efficiency of the Machine Learning model, so we are building the machine learning model with pipeline techniques and Hyperparameter Tuning technique. We are using the Twitter Training Dataset to train our model. Initial step is to do the Text pre-processing thereby removing the unwanted noise in the data. We are using the cross-validation method to overcome the underfitting and overfitting problems. We build the model with different classifiers, namely Logistic Regression, Decision Tree, Random forest, Multinomial Naive Bayes, Support Vector and vectorizers, namely count and term frequency-inverse document frequency (TFIDF). We tried the different combination of parameter tuning and found the best parameter which gives the maximum accuracy. After the analysis and voting classifier output, we found that the combination of TFIDF vectorizer and Random Forest Classifier pipeline gives the maximum accuracy of 96.57% and Predicts the tweets in real time as Positive or Negative.

S. Swamy (✉) · J. Kundale · D. Jadhav
Department of IT Engineering, RAIT, Sector 7, Nerul 400706, India

J. Kundale
e-mail: jyoti.jadhav@rait.ac.in

D. Jadhav
e-mail: dipti.jadhav@rait.ac.in

Keywords Sentiment analysis · Twitter · Pipeline · Hyperparameter tuning · Term frequency inverse document frequency voting classifier · Real-time tweets · Cross validation

1 Introduction

With the massive development in Digitization, more and more people are expressing their views, opinions, and feelings across the Internet and social media networks regarding the product, event, place, person, etc. This has increased the user-generated data containing Sensitive Information. Twitter is one of the small-scale blogging social media networking sites with millions of users. Sentiment of a text can be analyzed using Sentiment analysis which is also called Opinion Mining. Sentiment Analysis is a technique to know the user's opinion, emotions for a particular product, event, place, person, etc. Sentiment Analysis plays an important role in various fields like business, politics, sports, e-commerce, etc. [1]. With sentiment analysis, we can develop a more insightful, data-based marketing strategy. It helps to understand customers well. It is estimated that 80% of the data is unstructured and unorganized; a huge amount of text data is created everyday but hard to analyze, understand, and sort which is time-consuming and expensive. In such a case, Sentiment Analysis plays an important role in Business by Tagging the unstructured data. Sentiment Analysis helps in analyzing the Real-Time critical issues in Real Time. Reviews have significant impact on the success of any business [1]; positive reviews have positive impact on the success of a product while adverse reviews can affect the financial aspects [1]. So nowadays, most of the Business analysts prefer the sentiment analysis technique to analyze their product. A few uses of sentiment analysis are next selling opportunities, training chatbots, important emotional trigger identification, dealing with many customers, adjustable customer service, live insights, quick escalations, reduced customer churn, tracking overall customer satisfaction, and detecting changes in customer opinion; doing the analysis in real time can solve many critical real-time issues.

Here, we are doing the Sentiment Analysis of the Twitter Data which are mixed code, particularly English and Hindi. In India, millions of people communicate in the Hindi language. A major work is conducted on Sentiment Analysis of the English language, only fewer works are done in the multilingual sentiment analysis, as it needs a lot of preprocessing of data and resources [2]. We get many resources online like sentiment lexicon, and many data scientists create their own corpora and noise reducing algorithms, and then they call these text identification tools and apply the sentiment analysis model on them to check if the text is positive or negative [3].

The user-generated tweets are usually either in mixed-code, code-mixed, or in native script. Let us consider Example 1. The product I purchased is very good. I am satisfied with the product. Example 2. Product jo Maine karida Hai wo bahut accha hai, I am satisfied with the product. In the first example, the tweet is in only in English, whereas in the second, tweets are from different languages and it will be a

mixture of two languages but written in English script. Information access in these cases is very challenging as language identification is difficult and the transliteration needs lots of processing of raw data [4]. In recent times, research is conducted on tweets which are in a single language, but Twitter is a major microblogging service provider available online. So, there are multi-language tweets available daily and this must be analyzed for all the tweets to find the best sentiment of people. There is a need for a filter to opt out the useful information from the available vast amount of raw dataset that is fetched from the source. Then that filtered information can be used by organizations for analysis purpose which will help them to utilize the data to find out new emphasis. This will help the organizations and businesses to develop the smarter business, with productive operations, more profits, and happier customers [5].

The rest of the paper is organized as follow: Sect. 2 gives an overview of the literature review. In Sect. 3, the problems of the existing system of sentiment analysis of the multilingual data are discussed. In Sect. 4, the system is explained in detail and our main contribution in developing the model is also discussed. Experimental results are presented in Sect. 5 and a conclusion is provided in Sect. 6. For building the model, the authors have used the Twitter Training dataset, and the experiment is conducted on different Machine Learning algorithms, and the best parameters are selected based on the accuracy obtained. Our experimental results show a major improvement in classification accuracy and the model is more efficient.

2 Literature Review

Sentiment Analysis for Twitter data is done in many ways. Recent development in this field is using the Machine learning and Deep learning approaches.

Ansari and Govilkar [1] have done the Sentiment Analysis of the code-mixed data (Hindi, English, and Marathi languages). The authors have categorized the process into stages like language identification, word transliteration and sentiment score tagging, feature extraction. Then the authors have applied the Supervised Learning method to calculate the sentiment as output. The dataset used are 1200 Hindi and 3000 Marathi documents from social media networks from chats, tweets, and comments from YouTube, Facebook, and Twitter. They have analyzed the performance measure with Precision (false positive), Recall (false negative), and F1-score. They have achieved the accuracy of 90% for Marathi and 80% for Hindi.

Mishra et al. [6] have conducted the Sentiment Analysis process for the Hindi, Bengali, and English languages. In this approach, the authors have used Machine Learning and Neural Networks to achieve better accuracy for code-mixed data. They have used two models for Sentiment Analysis of code-mixed Hindi-English, Bengali-English datasets. The first model was an Ensemble Voting classifier, namely Linear SVM, Logistic Regression, and Random Forests, while the second model was Linear SVM. They have used the scikit-learn machine learning library for implementing both

approaches. They have used performance parameters Precision, Recall, and F1-score for comparison.

Vashistha et al. [7] have performed the Aggression Detection (it is a kind of Sentiment Analysis) for multilingual social media texts. They have worked on a combination of Hindi-English datasets. They have developed the model with features like word vectors, created manual dictionary of destructive words, sentiment scores, POS, and Emojis for classification tasks. They have used Machine learning and Deep learning models, and XGBoost classifier, Gradient Boosting classifier, and SVM are most suited for the task.

Shalini et al. [8] have conducted the sentiment analysis of Kannada-English code-mixed data. They first manually created the Kannada-English mixed Corpus by crawling Facebook comments; along with that they have used the sentiment analysis code-mixed Corpus available at Sentiment Analysis for Indian Languages (SAIL-2017). They have used the Distributed representation methods for Sentiment Analysis. They have achieved an accuracy for Hindi-English and Bengali English of 60.20% and 72.20%, respectively.

Yadav and Bhojane [9] have performed the Sentiment Analysis for Hindi and English code-mixed data. Here, the Hindi language words are in Devanagari script which are stored in UTF-8 encoding scheme. They have performed the sentiment analysis in 3 different approaches. Neural Network prediction using pre-classified words is used for the classification of data. IIT Bombay HindiSentiWordNet is used for classification. Classification is done by Neural Network prediction by using the pre-classified sentence as labeled data. They have calculated the accuracy for all the three above-mentioned approaches on different domain data. In the first method, they achieved an accuracy of 52%, and the second method 71.5% accuracy, and 70.27% accuracy in method 3.

Mukherjee [10] has proposed a Deep Learning technique for sentiment analysis of code-mixed English-Hindi text. They have used the Joint Learning Technique from character and word features for classification and used the Late fusion technique for analysis. The dataset was collected from public Facebook pages. They have chosen pages of famous Indian personalities like Salman Khan and Narendra Modi. The dataset contains 3879 comments. They have used Keros on Python for all required implementations. They have achieved the highest accuracy of 69.85%.

Pravalika et al. [11] have proposed a hybrid system for Sentiment Analysis for code-mixed English-Hindi Data. In the first approach, they have used the Lexicon, which is constructed using the sentiment present in the sentence. Sentiment combination rules are used to check the sentiment of a sentence. In the second approach, they have used the machine learning model which is built by using the mixed language training data. After extracting the matching features from the training set, the classifier has been trained to determine the polarity of the user comment. In the first approach, they have achieved an accuracy of 72%, and in the second approach an accuracy of 86%.

Bhargava et al. [12] have done the Sentiment Analysis of code-mixed data of the English language along with the regional languages like Hindi, Telugu, Tamil, and

Bengali. The whole process is split into two important steps. 1. Language Identification and 2. Sentiment Mining Techniques. The code-mixed sentence is transliterated to respective Indic script and is analyzed word by word for sentiment of the sentence. Dataset from FIRE 2015 is used for transliterated search which has information of eight Indian languages along with the English language. The precision, recall, F-measure, average F-measure, and weighted F-measure values are used as evaluation metrics to compare the performance. They have achieved a total accuracy of 67%.

3 Problem Statement

1. In most of the sentiment analysis processes, the input is English, Hindi, and code-mixed (English-Hindi), which are written in English letters. If the input contains the words in Devanagari script, then this existing system may not work properly. Our system can be used for all types of input data like Hindi words, English words, Hindi (Devanagari script), code-mixed (Hindi written in English script), and mixed-code words.
2. The major limitation of the existing system is that of Language Identification, if in case the Data is multilingual (contain words which are in more than 2 languages), then the existing system may not work properly. Our proposed system aims to overcome this limitation by using the Google Translator to convert any language text to respective English language text.
3. The existing system does not use any Pipeline method; thus, the whole process is time-consuming while performing the Sentiment Analysis. Our system uses the Pipeline concept to overcome this limitation. The processing time and prediction time are less compared to other sentiment analysis techniques.
4. In the existing systems, they have used the Translation dictionaries which are available online; this may not work properly for all words if different languages are used simultaneously. Imagine if the same words are used at different contexts then there is a need for context-dependent mapping of the word which is a big task and not 100% accurate and usually are error prone, and manually need to analyze it. In our system, we are using the Google Translator which overcomes this limitation.
5. In this existing system, the accuracy is about 80%, whereas our proposed system has an accuracy of more than 95% as we are using the hyperparameter tuning technique to improve the accuracy of the model.
6. The existing system does not have any cross-validation process. Our system uses the cross-validation technique to validate the stability of the models.

4 Proposed System

Our proposed system aims to overcome all the limitations mentioned in the previous section. The main challenge was to increase the accuracy of the model. The system is divided into these major steps (Fig. 1).

4.1 Fetch the Twitter Training Dataset

Training data is the initial data that we use to develop an efficient machine learning model, from which the model refines and creates its rules to train the model; we need to have a high-quality training dataset which we are importing from Twitter for our implementation. We can fetch these training data usinf online and offline modes. (Twitter-Sentiment-analysis/master/train.csv) https://raw.githubusercontent.com/dD2405/Twitter_Sentiment_Analysis/master/train.csv.

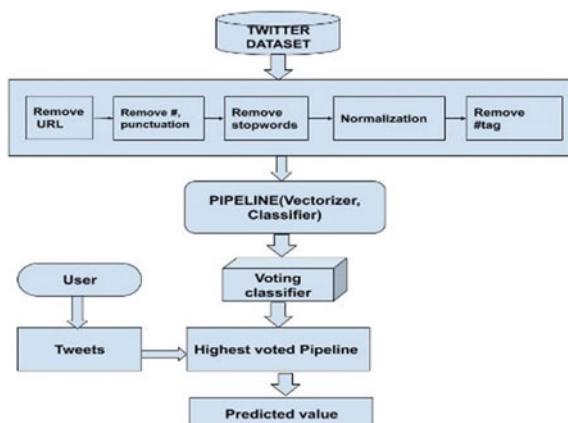
4.2 Text Pre-processing

Text pre-processing is a major process in the sentiment analysis of Twitter data. The following steps are performed in text pre-processing.

4.2.1 Remove URL

In this step, we are creating a function to remove the URLs present in the training data as these URLs do not add any value to the sentiment analysis process.

Fig. 1 Proposed system of sentiment analysis of Twitter data using machine learning model



4.2.2 Remove Punctuation, Special Characters, and Numbers

The special characters, numbers (12.43, etc.), and punctuation do not have any impact on the sentiment analysis process. We are creating the function to remove it.

Remove stop words, alpha numerical characters.

In many cases, we find that there are many stop words and alpha numerical characters used in the documents which are not contributing anything for the sentiment analysis process. So, we are creating a function to remove the same.

Text Normalization.

In Normalization, we are converting text into a more convenient, standard form before performing any other function for higher level modeling. Lemmatization process is done here to convert the words by comparing the word with vocabulary and morphologically analyzing the words. Usually, the target is to identify the inflectional endings, and the base or dictionary form of a word, known as the lemma, is returned.

Remove Two-letter words.

In this stage, the function is created to remove the two-letter words like an, at, is, to, in, etc. These two-letter words do not add any value to the analysis, rather they increase the count values in the vectors. The two-letter word-removed tweets are returned for further analysis.

4.2.3 Applying Text Pre-processing to the Training Dataset

The new tweet dataset is created by applying Text pre-processing to the training dataset. The new Training dataset is free from all punctuation, numbers, special characters, alpha numerical characters, URLs, stop words, etc.

4.2.4 Import Word Cloud, Image Color Generator

Here, we are importing the word cloud, Image Color Generator, to display all positive and negative words from the Training dataset and display the same using the plot. From the figures, we can show that the maximum highlighted words have occurred more in the tweets.

4.2.5 Hashtag-Append Function

Hashtag plays an important role in the analysis of Sentiment, as it helps to organize and sort certain types of tweets. Hashtags are usually created by merging several

words together. Splitting hashtag or other words mistakenly joined becomes handy especially when we try to run Sentiment Analysis.

This function is created to check and display any hashtag present in the data. Hashtag-connected positive and negative words in the data is fetched and the same is displayed using the bar graph. From the bar graph, we can see how many hashtag positive and negative words are present in the document. And the same is stored in the list.

4.2.6 Function to Remove the Hashtag

This function is created to remove the hashtag symbol in the data, as this symbol is not of much importance in the sentiment analysis process. The new list is created after removing the hashtag.

4.2.7 Train Test Split Function

Here in our project, we are using a single dataset, hence we are splitting it by using the Sklearn train test split function first. The train test split function provides the provision to split the dataset to achieve the ideal model. Also, the model should not be overfitting or underfitting, so we are doing the cross validation.

4.2.8 Machine Learning Pipelines

Generally, while working on the Machine Learning algorithm, we work on a series of tasks like pre-processing, feature extraction, model fitting, and validation stages. With large or huge amount of datasets, using as many libraries as possible for each stage is not an easy task. Many of the Machine Learning libraries are not suitable to work for distributed computations, or many do not provide full co-operation for hyperparameter tuning and pipeline creation. The Machine Learning Pipelines are high-level API for Machine Learning libraries that work under the spark.ml package. A pipeline consists of a series of stages. There are two general types of pipeline stages.

1. Transformer

For Transformer, augmented dataset is obtained as output if it is sent as input. Example: Tokenizer is a transformer that transforms a given dataset with text into a dataset with tokenizer words. In our project, we are creating the pipelines using the two types of vectorizers, namely 1. Count vectorizers and 2. TFIDF vectorizers.

2. Estimators

Initially, the input dataset is fit on Estimator to create the model, where the transformer transform available input dataset. Example: Random Forest classifier is an estimator

which gets trained on a dataset along with labels and features and creates a Random Forest model.

In our project, we are creating Estimators with Logistic Regression, Decision Tree classifiers, Random Forest classifier, Multinomial Naive Bayes classifier, and Support Vector classifier. Here in our project, we are using both count and TFIDF vectorizers in combination with all the above-mentioned classifiers. The pipeline is created using first the TFIDF vectorizer (stop words are set to English) and Logistic Regression (random state is set to 0, solver as liblinear) with all default values. The accuracy is calculated with all Default values. We even calculate the classification report like Recall, Precision, F1-score, Support, and Confusion matrix along with the Training time and Prediction time.

4.2.9 Hyperparameter Tuning

The Hyperparameter Tuning is done

To determine the Optimal values for a given model by finding the best combination of parameters for specific problem analysis.

To increase the performance of the model.

To lower the ERROR rate.

To reduce the Processing time.

To maximize the Accuracy of the model.

Here, we are performing the Hyperparameter tuning to increase the accuracy of the model. We performed both types of Hyperparameter tuning, namely GridsearchCV and RandomsearchCV. From the analysis, we found that Random search CV has relatively lesser run time compared to GridsearchCV. And the performance was also the same for both the tuning techniques. So, we are using RandomsearchCV for further analysis. We are creating the pipeline with n-jobs, n-iter, random state, and CV as the parameters for initializing the tuning. Different parameters are passed to the tuning technique and optimized results are obtained.

4.2.10 Voting Classifier

Voting Classifier is used in building the efficient machine learning model which gets trained on an Ensemble of different models. Ensemble is nothing but an assembling of different models which are sharing a single dataset together. The main advantage of using the Ensemble is to reduce the error and lower the overfitting problems. Voting classifier is used when we are working on different machine learning algorithm to combine the prediction. Voting classifier is not a classifier but a folder for a set of different classifiers. Here, we are passing the different pipelines created with the vectorizers and classifiers as Estimators for Voting Classifier, namely Logistic Regression, Decision Tree, Random Forest, Multinomial Naive Bayes, and Support Vector classifiers. The output obtained are tabulated and a graph is plotted.

4.2.11 Mixed-Code Twitter Data

Our aim is to do the sentiment analysis of mixed-code Twitter data. To achieve this, we must access the real-time Twitter data. To do so, we need to have a Twitter active account, and have the authority to develop access and create own application which will create API credentials. The API credentials are needed to access Twitter's tweet through Python. Now to access the Twitter API, we need four keys, namely consumer key, consumer secret key, access token key, and access token secret key. After the authentication, we can access the tweets and use them for further analysis. Here, we are accessing the mixed-code tweets and translating them to English using Google Translator, and do the Text pre-processing. As we got maximum accuracy in TfidfVectorizer and RandomForestClassifier, we call this model for prediction. Our model is giving the accuracy of 96.57%.

5 Results and Analysis

To run the codes, we are using Jupyter Notebook which is an open-source web developing software. Jupyter Notebook is used by Data scientists for developing data science projects. We use Python (version 3.9) language. Most of the libraries used for machine learning have Python interfaces. We are using Jupyter Notebook to create the Machine Learning model and run the codes. We have collected the data from social media network, Twitter. To collect the data, we use Twitter API, i.e., tweepy master. We are using scikit-learn library to run the Machine Learning algorithms. Also, we are using NLTK library to work with natural languages and Google Translator to convert the Natural Language to English. • Python libraries: Re: Regular Expression library, which gives regular expression matching operations. Pandas: Software Library used for Python programming language for Data manipulation and analysis. NumPy: Library used while working with arrays. It has functions while working in domains like Linear algebra, Fourier transform, etc. Matplotlib: Plotting library for Python programming language. An object-oriented API is given for combining plots along with applications with the help of general-purpose GUI toolkit. NLTK: Natural language toolkit used for natural language processing for the English language written in Python programming language. Googletrans: It is a module used to translate the Text. Sklearn: sci-kit learn is a Machine Learning library for Python, and it features many algorithms like SVM, logistic regression, random forest, etc. It also supports Python numerical and scientific libraries like pandas and NumPy. The following is the results obtained for our experiment. The dataset for training is accessed from (https://raw.githubusercontent.com/dD2405/Twitter_Sentiment_Analysis/master/train.csv).

Hashtag plays an important role in Sentiment Analysis; we count the hashtags positive and negative words as shown in the following graphs (Fig. 2).

We are creating the machine learning model by the pipeline technique; we have done the cross validation using Random Search CV and Grid Search CV. From the

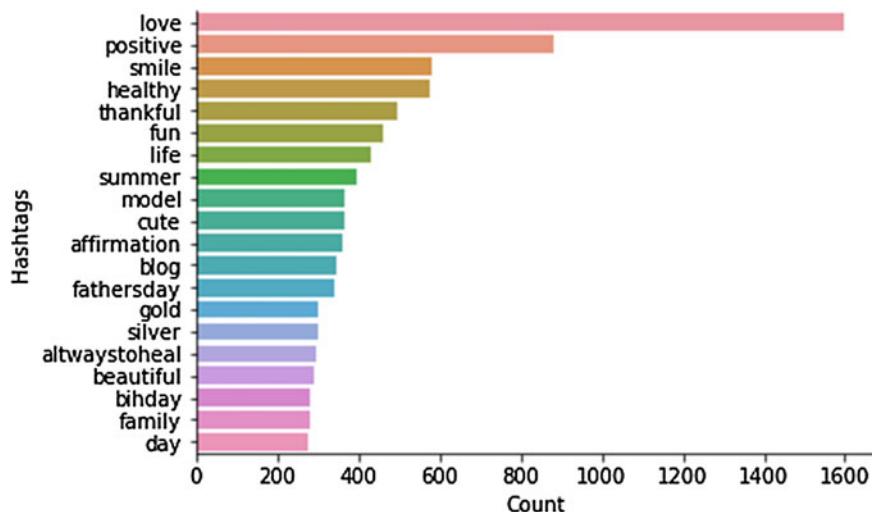


Fig. 2 Graph showing all the hashtagged positive word count

analysis and output, we found that the processing and training times for RandomsearchCV are relatively less compared to grid search CV and the accuracy score for both the Cross-Validation technique is the same. So, we are using RandomsearchCV for further analysis (Fig. 3).

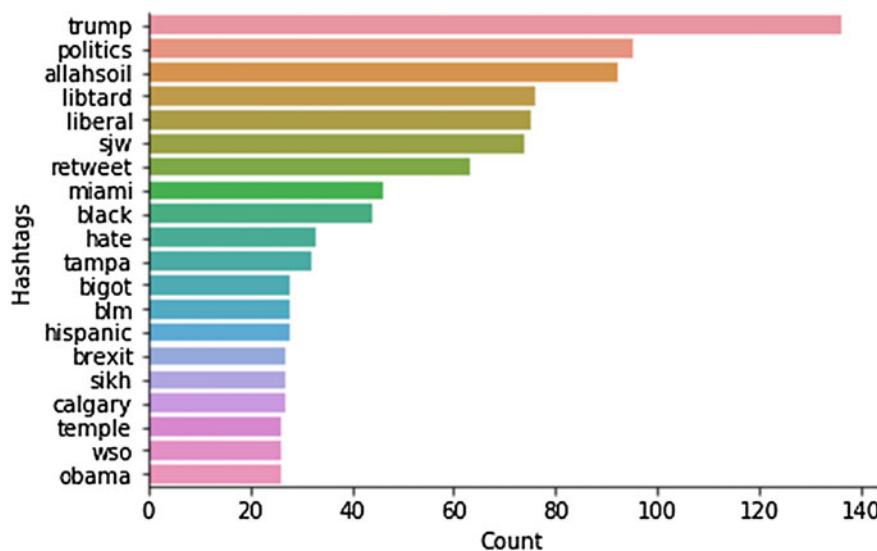


Fig. 3 Graph showing all hashtagged negative word count

From the following tables, we see that the voting classifier is giving the highest vote to Random Forest classifier, and the same is plotted in the graph (Table 1 and Fig. 4).

Here, the same process is repeated with count vectorizers, and the output is tabulated. From the voting classifier, we see that maximum vote is given to the Random classifier (Table 2 and Fig. 5).

Table 1 Comparison table of different classifiers and TFIDF vectorizer with accuracy evaluating parameters

Model	Logistic regression	Decision tree	Random forest	Multinomial NB	SVC	Voting
Vectorizer	TFIDF	TFIDF	TFIDF	TFIDF	TFIDF	TFIDF
Accuracy score	0.96199	0.94885	0.965431	0.950884	0.962303	0.965431
Training time	0.74884	14.2862	20.3583	0.797074	93.484	112.109
Predicted time	0.10032	0.100307	1.43972	0.120984	10.4484	7.47972

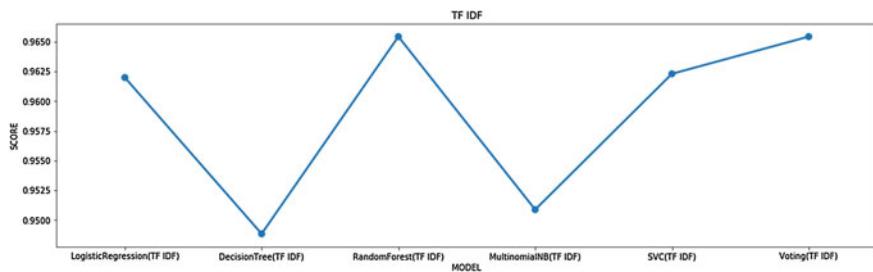


Fig. 4 Comparison graph of different classifier (TFIDF) with accuracy score

Table 2 Comparison table of different classifiers and count vectorizer with accuracy evaluating parameters

Model	Logistic regression	Decision tree	Random forest	Multinomial NB	SVC	Voting
Vectorizer	Count	Count	Count	Count	Count	Count
Accuracy score	0.963867	0.956984	0.967308	0.964023	0.962303	0.965431
Training time	2.12536	7.23073	25.6612	0.438491	218.721	238.022
Predicted time	0.321061	0.0808282	1.55258	0.110948	11.1043	12.6393

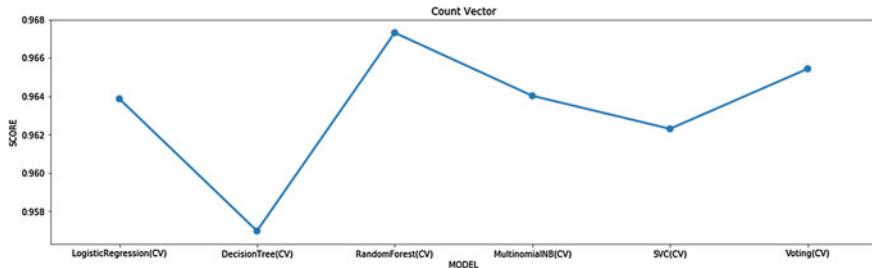


Fig. 5 Comparison graph of different classifiers (count) with accuracy score

Table 3 Comparison table of evaluating parameters of existing system and proposed system

Parameters	Existing system [6]	Proposed system
Algorithm used	SVM with TFIDF vectorizer	Random forest with TFIDF vectorizer
Precision	0.6	1.0
Recall	0.58	0.97
F1-score	0.58	0.98
Accuracy (%)	83.7	96.57

From both the tables and performance evaluating parameters, we conclude that the TFIDF vectorizer and Random Forest classifier are giving the maximum output of about 96.57%. After developing the efficient model, we are fetching the Twitter data for prediction using the tweepy API. As mentioned earlier, we are giving the input (mixed code), namely Hindi and English language tweets and predict the output as positive or negative (Table 3).

6 Conclusion and Future Scope

The Sentiment Analysis of Twitter data is performed by building the machine learning model. We conducted the sentiment analysis of Twitter data which are mixed code (Hindi and English) in real time and found out whether the tweet is Positive or Negative. We developed a more accurate and efficient machine learning model by using the automated process called the Pipeline technique. We tried different combinations of the classifiers and vectorizers while creating the pipeline. This pipeline technique has helped us achieve simplified, efficient, and run time optimization model. Our main aim was to improve the accuracy of the model which we achieved by hyperparameter tuning technique which is our main contribution in developing the efficient model. We tried all possible combinations of hyperparameter tuning and found the best parameter combination which gives the Maximum accuracy. Our model is more accurate compared to existing models present. We conducted the experiment

with both Count and TFIDF vectorizers and different classifiers and found out that TFIDF vectorizer and Random Forest classifier give the maximum accuracy with almost 96.57% with less processing and predicting times. The future scope of the project is to do sentiment analysis on different social media networks like Facebook comments, YouTube comments, Amazon customer reviews, etc. In our paper, we are doing sentiment analysis of text data; in future, we can extend our sentiment analysis process for the image and emojis and gif also. Future scope of the project is extending the system for detecting any spam tweets before doing the sentiment analysis. And we can extend our project for big data analysis.

References

1. M.A. Ansari, S. Govilkar, Sentiment analysis of mixed code for the transliterated Hindi and Marathi texts. *Int. J. Nat. Lang. Comput. (IJNLC)* **7**(2) (2018)
2. A. Poornima, K. Sathiya Priya, A comparative sentiment analysis of sentence embedding using machine learning technique, in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)* (2020)
3. D.K. Prabhakar, S. Pal, Machine transliteration and transliterated text retrieval: a survey. *Sadhana* **43**, 93 (2018) (Indian Academy of Sciences). <https://doi.org/10.1007/s12046-018-0828-8>
4. A. Hasan, S. Moin, A. Karim, S. Shamshirband, Machine learning based sentiment analysis for Twitter accounts. *Math. Comput. Appl.* **23**, 11 (2018). <https://doi.org/10.3390/mca23010011>, www.mdpi.com/journal/mca
5. A.K. Soni, *Multi-Lingual Sentiment Analysis of Twitter Data by Using Classification Algorithms* (2017). ISBN: 978-1-5090-3239-6/17/31.00©2017IEEE
6. P. Mishra, P. Danda, P. Dhakras, *Code-Mixed Sentiment Analysis Using Machine Learning and Neural Network Approaches*. Language Technologies Research Center, IIIT Hyderabad, Kohli Center on Intelligent Systems (2018)
7. N. Vashistha, A. Zubiaga, Aggression detection on multilingual social media text, in *10th ICCCNT-2019*, IIT-Kanpur, Kanpur, India, Nov 2020
8. K. Shalini, H.B. Barathi Ganesh, M. Anand Kumar, K. Soman, *Sentiment Analysis for Code-Mixed Indian Social Media Text with Distributed Representation* (IEEE, 2018). ISBN: 978-1-5386-5314-2/18/2018
9. M. Yadav, V. Bhojane, *Semi-Supervised Mix-Hindi Sentiment Analysis Using Neural Network* (2019). ISBN: 978-1-5386-5933-5/19/2019
10. S. Mukherjee, *Deep Learning Technique for Sentiment Analysis of Hindi-English Code-Mixed Text using Late Fusion of Character and Word Features* (IEEE, 2019). ISBN: 978-1-7281-2327-1/19
11. A. Pravalika, V. Oza, N.P. Meghana, S. Sowmya Kamath, Domain-specific sentiment analysis approaches for code-mixed social network data, in *8th ICCCNT 2017*, IIT Delhi
12. R. Bhargava, Y. Sharma, S. Sharma, Sentiment analysis for mixed script Indic sentence, in *Conference on Advances in Computing, Communications, and Informatics (ICACCI)*, Jaipur, India, 21–24 Sept 2016
13. C. Nanda, M. Dua, G. Nanda, Sentiment analysis of movie reviews in Hindi language using machine learning, in *International Conference on Communication and Signal Processing*, India, 3–5 Apr 2018
14. K. Shalini, A. Ravikumar, R.C. Vineetha, D. Aravinda Reddy, M. Anand Kumar, K.P. Soman, Sentiment analysis of Indian languages using convolutional neural networks, in *2018 International Conference on Computer Communication and Informatics (ICCCI-2018)*, Coimbatore, India, 04–06 Jan 2018

15. M. Singh, V. Goyal, S. Raj, Sentiment analysis of English-Punjabi code mixed social media content for agriculture domain, in *2019 4th International Conference on Information Systems and Computer Networks (ISCON)*, GLA University, Mathura, UP, India, 21–22 Nov 2019
16. M. Wongkar, A. Angdresey, Sentiment Analysis Using Naive Bayes Algorithm of the Data Crawler: Twitter. Authorized licensed use limited to: University of Birmingham. Downloaded on 11 May 2020 at 12:39:07 UTC from IEEE Xplore
17. S. Tiwari, A. Verma, P. Garg, D. Bansal, Social media sentiment analysis on Twitter datasets, in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*
18. V.A. Kharde, S.S. Sonawane, Sentiment analysis of Twitter data: a survey of techniques. *Int. J. Comput. Appl.* **139**(11) (2018). ISSN: 0975-8887
19. S.C. Rachiraju, M. Revanth, Feature extraction and classification of movie reviews using advanced machine learning models, in *Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020)*. IEEE Xplore Part Number: CFP20K74-ART; ISBN: 978-1-7281-4876-2
20. N. Jose, B.R. Chakravarthi, S. Suryawanshi, E. Sherly, J.P. McCrae, A survey of current datasets for code-switching research, in *2020 6th International Conference on Advance Computing and Communications Systems (ICACCS)*
21. S. Dhawan, K. Singh, P. Chauhan, Sentiment analysis of Twitter data in online social network, in *5th IEEE International Conference on Signal Processing, Computing and Control (ISPCC 2k19)*, JUIT, Solan, India, 10–12 Oct 2019
22. R. Wagh, P. Punde, Survey on sentiment analysis using Twitter dataset, in *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA 2018)*. IEEE Conference Record 42487; IEEE Xplore ISBN: 978-1-5386-0965-1
23. S.V. Pawar, S. Mali, Sentiment analysis in Marathi language. *Int. J. Recent Innov. Trends Comput. Commun.* ISSN: 2321-8169
24. V. Goel, A.Kr. Gupta, N. Kumar, Sentiment analysis of multilingual Twitter data using natural language processing, in *2018 8th International Conference on Communication Systems and Network Technologies*. ISBN: 978-1-5386-5956-4/18
25. M.H. Abd El-Jawad, R. Hodhod, Y.M.K. Omar, Sentiment analysis of social media networks using machine learning, in *14th International Computer Engineering Conference (ICENCO)* (IEEE, 2019)
26. S. Kamiş, D. Gouliaras, Evaluation of deep learning techniques in sentiment analysis from Twitter data, in *2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)*
27. J.A. Alzubi, A. Kumar, O.A. Alzubi, R. Manikandan, Efficient approaches for prediction of brain tumor using machine learning techniques. *Indian J. Public Health Res. Dev.* (2019). <https://doi.org/10.5958/0976-5506.2019.00298.5>
28. O.A. Alzubi, J.A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, M. Ramachandran, An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* (2020). <https://doi.org/10.1007/s00521-020-04761-6>

Residual Decoder based U-Net for Semantic Segmentation



Shilpa Elsa Abraham and Binsu C. Kovoor

Abstract Semantic segmentation is the preliminary step in several computer vision tasks and therefore is a hot topic of research. Convolutional Neural Networks have invaded the domain by their representation power and proved their efficacy in the recent years. This paper proposes an encoder–decoder-based segmentation model, with skip connections established across encoder and decoder blocks in a U-shape. When most of the existing models focus on the encoder part, the proposed model attempts to enhance the decoder module to recover the spatial information effectively using residual connection. Pretrained VGG16 is employed to capture the high-level features of the image via the encoder module. In the proposed decoder module, two variants of convolutional blocks with residual connection are designed, in which one uses serial convolutional layers and the other uses parallel convolutional layers. The use of dilated convolutions in an optimal manner improved the performance further. Experimental analysis on a dataset of road images shows that the model outperforms the basic U-net considerably, with a mIoU of 77.86% and 76.74% for serial and parallel variants, respectively. It shows promising performance in separating the semantically relevant objects in the image. Despite using convolutions with large-sized kernels, the model is able to be trained with less number of parameters with the proposed parallel variant of the residual decoder-based model.

Keywords Semantic segmentation · Deep learning · Convolutional neural network · Residual connection · VGG16 · Atrous convolution · Encoder–Decoder

1 Introduction

The profound success of machine learning [1, 2] and deep learning techniques, especially convolutional neural network (CNN), has revolutionized the research in the field of computer vision. The CNNs, originally introduced by image classification researchers [3–5], utilized their feature extraction capability, soon got adopted by

S. E. Abraham · B. C. Kovoor (✉)

Division of Information Technology, Cochin University of Science and Technology, Kochi, Kerala, India

other domains involving structured prediction. They have turned out to be the dominating driving force of several computer vision tasks including object detection [6, 7], object localization, saliency detection [8], semantic segmentation [9–11] and so on. Semantic Segmentation is the task of separating each object in the image based on its class. It is a combined task involving classification, object detection and localization. It has proved its applications in a variety of domains like self-driving cars [12], scene parsing [13] and medical image diagnostics [14, 15].

With the advent of deep learning, several deep convolutional neural networks [4, 5] captured the research community and outperformed the existing approaches that rely on handcrafted features. Fully Convolutional Networks (FCN) [11] turned out to be a very promising approach for end-to-end semantic segmentation. Following the idea from FCN, several encoder–decoder structures were proposed that proved to be simple, yet effective, for pixelwise prediction. Segnet [16] and U-net [14] are powerful encoder–decoder models. Spatial Pyramid Pooling Network (SPP-Net) [17] is yet another significant approach that let different-sized input images to be fed into CNN architecture. Apparently, most of them focused on the feature extraction part and blindly ignored the reconstruction part, which constituted segmentation mask prediction. U-net put forward by Ronneberger et al. [14], though originally built for medical image segmentation, has now been tried by other domains and its excellence is proved [18, 19]. In this paper, a structure similar to that of the U-net with improved decoder architecture is modelled. Decoder segment is primarily designed with residual-based convolutional blocks. Two versions of the model are proposed, which comprise serial convolutional layers and parallel convolutional layers, respectively.

VGG16 [4], originally built as an image classification model, as part of ILSVRC-2014 has been widely used as the backbone network in several deep learning-based computer vision tasks. This aids in reduced training time and generalization error for the new model. The VGG model’s ability to extract the best features integrated with new models has produced significant results in the past. In the proposed model, pretrained VGG16 is fed as the encoder part. Particularly, the success of VGG16 in correctly classifying the pixels in the image is best leveraged in the model to extract the semantically rich information from the image.

The rest of the paper is organized as follows. Section 2 describes the proposed methodology with encoder–decoder module architectures explained. It also details information on model training and metrics used for the evaluation. Section 3 discusses the results obtained and ablation study followed by conclusion in Sect. 4.

2 Proposed Methodology

The proposed architecture is an encoder–decoder structure, which extracts multidimensional features from the input image and then reconstructs the object segmentation from the extracted features. The output segmentation map will be of the same dimension as that of the input image, thus allowing easy analysis. Figure 1 illustrates

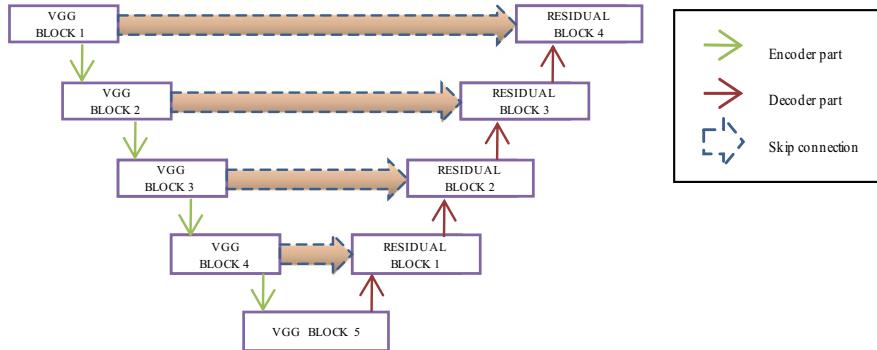


Fig. 1 Basic framework

the basic framework of the model. It consists of a series of convolutional blocks in the contracting path (encoder module) that best extracts the high-level features of the input, followed by a series of residual blocks in the expanding path (decoder module) to predict the object mask.

2.1 Encoder–Decoder Module Architecture

The primary aim of the encoder network is to extract the robust features of the image, from which the segmentation mask can be reconstructed. The model uses pretrained VGG16 [4] as the encoder part, which has the ability to understand the pixel-level features very accurately. Since the VGG network requires input images of size (224, 224), original images of size (360, 480) have been resized to the required dimension. The pixel values have undergone scaling also so as to be fed into the VGG16. To suit the image classification model for the segmentation task, the fully connected layers and softmax layer in the VGG have been dropped so as to retain the high-resolution feature maps, which are to be forwarded to the decoder module.

The model also employs skip connections in the network from the encoder segment to the decoder segment in a symmetric manner. These connections are introduced in a similar manner as that of U-net [14]. This enables the spatial information, which must have been lost during the max pooling operations in the encoder part to propagate to the decoder part. Hence, features from different resolution levels are merged, which proved to be very efficient in the better construction of segmentation maps.

The decoder module primarily tries to build the segmentation mask from the extracted features. In order to maintain the U structure, the model is equipped with four decoder blocks as shown in Fig. 1. In the proposed architecture, the decoder part is designed in a twofold approach. Figure 2 illustrates the design of the proposed residual decoder module.

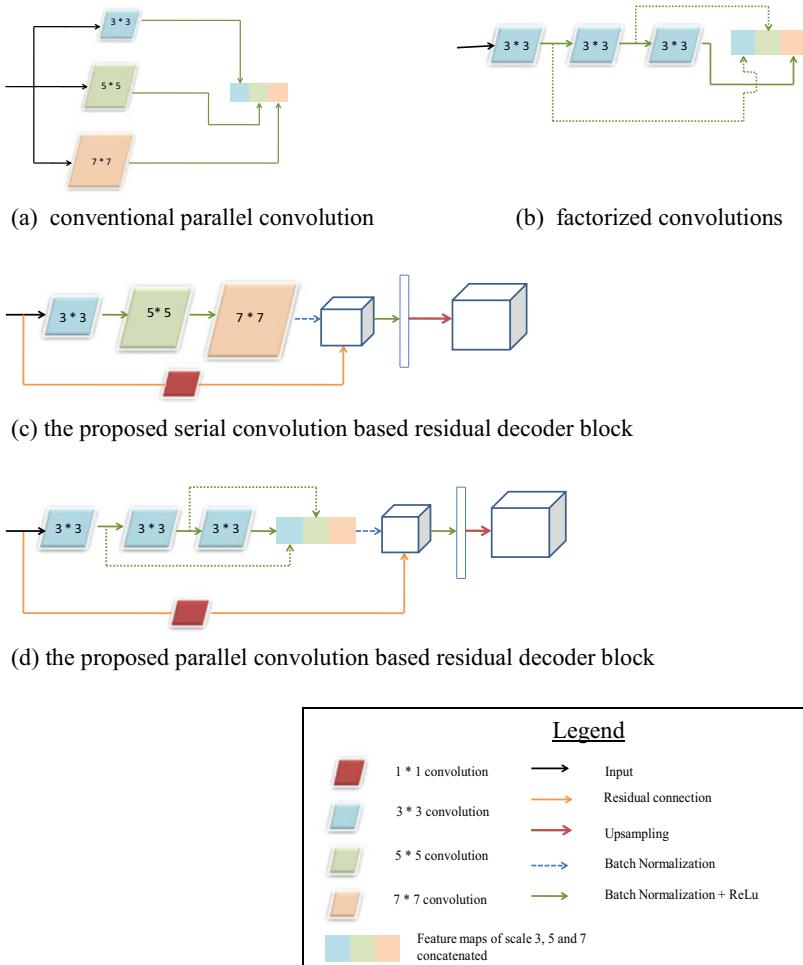


Fig. 2 Design of residual decoder block architecture

As represented in Fig. 2c, the first approach utilizes a series of three convolutional layers of $(3, 3)$, $(5, 5)$ and $(7, 7)$ kernel sizes, respectively, in each block. The increased filter size in successive convolutional layers enables capturing of details at different scales from the feature maps. The residual connection is made along a 1×1 convolutional layer, which comprehends some more spatial information. This connection negates the possibility for any vanishing gradient problem that may affect the prediction. The feature map resulting from 7×7 convolution undergoes batch normalization and gets concatenated with the residual connection, which is then applied with the Rectified Linear Unit (ReLU) activation function and batch normalization. This is then upsampled to be fused with the skip connection from the corresponding encoder block.

In the second approach, in lieu of employing a series of convolutional layers, parallel convolutional layers of (3, 3), (5, 5) and (7, 7) kernel sizes (Fig. 2d) are applied in each residual block. This enables to incorporate spatial information from different context sizes. Rather than applying in a conventional manner (Fig. 2a), the proposed model implements parallel convolutions in a factorized manner [20], which reduces the computation time drastically. A 5 * 5 convolution can be realized using a series of two 3 * 3 convolutions, and a 7 * 7 convolution using a sequence of three 3 * 3 convolutions. This is pictorially illustrated in Fig. 2b. Accordingly, the proposed model uses factorized 5 * 5 and 7 * 7 convolutions. The model accomplishes this by a series of three 3 * 3 convolutional layers, wherein the output of the second 3 * 3 convolutional layer resembles a 5 * 5 convolution, and the output of the third convolutional layer resembles a 7 * 7 convolution. Hence, the outputs of these three convolutional layers can be concatenated in order to get the effect of parallel convolution on the input feature map. This undergoes batch normalization and before applying the activation function, the feature maps are concatenated with the residual connection along a 1 * 1 convolutional layer. This is then followed by ReLu activation, batch normalization and finally upsampling.

The first residual block of the decoder segment makes use of atrous convolution [21]. This paved way for better detail exploration from the extracted high-level features, which laid a strong foundation for precise localization. It allows the semantically rich feature maps to be analysed more broadly and reconcile the information so as to achieve greater performance without increasing computational cost.

2.2 Model Training

The dataset comprises images of road scenes, which contain buildings, cars, lanes and everything around a road scene. Each image may contain up to 12 classes, including the background. The dataset is so disparate that the images contain objects of the same class in different resolutions in multiple images.

The experiments are conducted using Python 3 programming language, in which the models are implemented in Keras using TensorFlow backend. In order to optimize the proposed segmentation model, Adam optimizer and cross-entropy loss function are used during the training process.

2.3 Evaluation Metrics

Three metrics have been used in order to evaluate the proposed segmentation model. The following notations are used in the metrics: k denotes the total number of classes, and p_{ij} denotes the number of pixels of class i , predicted to be of class j . Hence, p_{ii} indicates true positive(TP), p_{ij} and p_{ji} representing false positive (FP) and false negative (FN), respectively.

- **Pixel Accuracy (PA):** This measures the number of pixels that have been classified correctly in the prediction map as shown in Eq. (1).

$$\text{PA} = \frac{\sum_{i=1}^k p_{ii}}{\sum_{i=1}^k \sum_{j=1}^k p_{ij}} \quad (1)$$

- **Mean Intersection over Union (mIoU):** Intersection over Union is the ratio of ground truth (intersection) and prediction (union) as given in Eq. (2). This metric calculated over a single class is averaged across all the classes in order to obtain mIoU (Eq. (3)):

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \quad (2)$$

$$\text{mIoU} = \frac{1}{k} \sum_{i=1}^k \frac{p_{ii}}{\sum_{j=1}^k p_{ij} + \sum_{j=1}^k p_{ji} - p_{ii}} \quad (3)$$

- **Dice coefficient (DC):** Dice coefficient (for a single class) is the ratio of double the intersection and sum of union and intersection. This is again calculated over a single class and averaged across multiple classes so as to obtain final Dice coefficient, calculated using Eq. (4):

$$\text{DC} = \frac{1}{k} \sum_{i=1}^k \frac{2 * p_{ii}}{\sum_{j=1}^k p_{ii} + \sum_{j=1}^k p_{ij} + \sum_{j=1}^k p_{ji} - p_{ii}} \quad (4)$$

3 Results and Discussion

To evaluate the performance of the model, 5 iterations of experiments have been conducted on the dataset. In each iteration, the accuracy obtained is recorded and then averaged in order to obtain the final accuracy.

As shown in Fig. 3a, the model performs exceptionally well with mIoU of 76.74% and 77.86% in the serial and parallel variants, respectively. Accordingly, Dice coefficient and pixel accuracy also excel (Fig. 3b, c). The model is compared with two more semantic segmentation models, wherein significant performance gain has been achieved in contrast to basic U-net (indicated as B U-net).

Also, an encoder-decoder-based semantic segmentation model with VGG16 encoder and a symmetric decoder is implemented, for which mIoU is obtained as 74.53%. This architecture is named Sym-Dec in the figure. Figure 3 illustrates the details of results obtained. All the architectures are implemented with similar skip connections between corresponding encoder and decoder blocks.

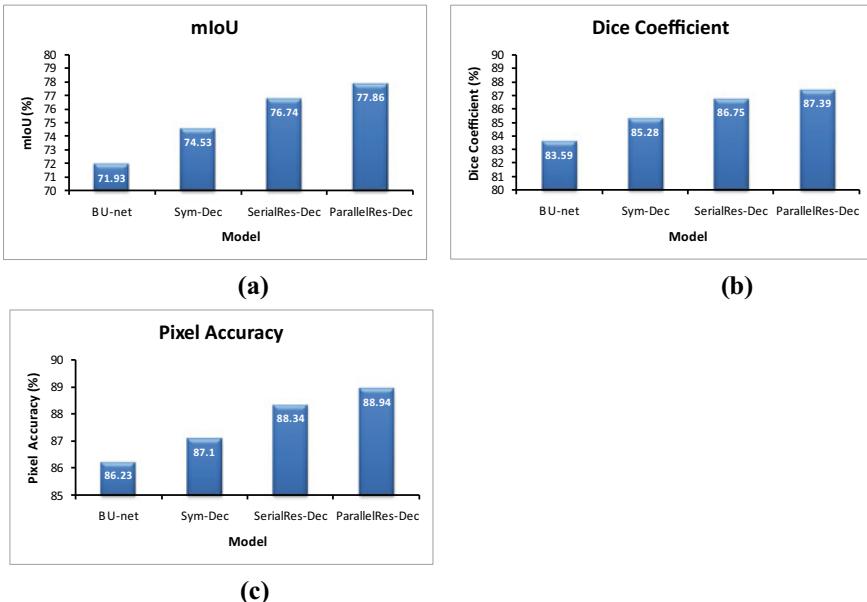


Fig. 3 Performance of different architectures in terms of **a** mIoU, **b** Dice coefficient and **c** pixel accuracy

Ablation study with respect to atrous convolution is also conducted. The results in Tables 1 and 2 demonstrate the effect of atrous convolution. The experiments conducted revealed that the serial convolution-based residual architecture provided best results when dilation rate is set to 2 and parallel convolution-based model starred at the rate of 4. The model is formulated with atrous convolution in the first residual block of the decoder part. This is because the high-level, semantic features of the

Table 1 Effect of atrous convolution on the serial convolution-based residual decoder

	mIoU (in %)	Dice coefficient (in %)	Pixel accuracy (in %)
No dilation	74.58	85.30	87.35
Dilation rate of 2	76.74	86.75	88.34
Dilation rate of 4	76.22	86.30	88.02
Dilation rate of 6	72.90	84.20	85.85
Dilation rate of 8	70.67	82.58	84.62

Table 2 Effect of atrous convolution on the parallel convolution-based residual decoder

	mIoU (in %)	Dice coefficient (in %)	Pixel accuracy (in %)
No dilation	74.86	85.39	87.42
Dilation rate of 2	76.98	86.85	88.50
Dilation rate of 4	77.86	87.39	88.94
Dilation rate of 6	75.75	86.02	87.75
Dilation rate of 8	73.62	84.67	86.00

image are best captured in the last feature map of the encoder block and can be best grasped in the very next block.

Though the model is designed twofolds the parallel convolution-based residual decoder approach has a higher advantage of fewer numbers of parameters to be trained. It is observed that, using the parallel approach, the number of parameters trained is limited to 24 M, whereas for the serial approach, it turned out to be 40 M, though the network seemed so simple.

Figure 4 illustrates a few visualization results on the dataset with the proposed model applied. When Fig. 4a, b illustrates original image and ground truth, Fig. 4c–e shows the predicted segmentation mask for the proposed model (ParallelRes-Dec), Sym-Dec and U-net, respectively. The model is found to provide satisfactory segmentation with very clear boundaries for larger objects and fairly decent outlines for distant objects even. As visualized in Fig. 4c, the proposed model is able to clearly differentiate the person on the road unlike other models in Fig. 4d, e. Also, the model is able to attain smooth quality segmentation mask.

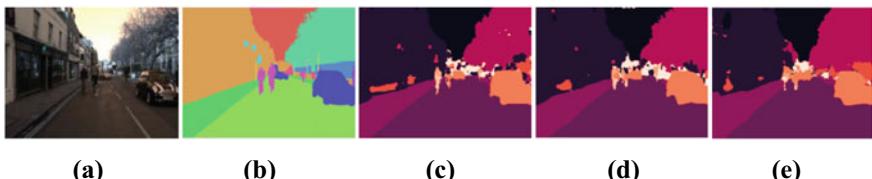


Fig. 4 Visualization results: **a** original image, **b** ground truth, **c** ParallelRes-Dec (proposed model), **d** Sym-Dec and **e** B U-net

4 Conclusion

The proposed model presents a novel architecture for the high-demanding semantic segmentation task. In this work, an encoder–decoder-based model is structured which is simple, yet effective. The very successful pretrained VGG16 is reused in the contracting path to capture the context of the image. The high-level semantic features captured by this module are fed into the decoder part to recover the image as a segmentation mask efficiently. The expanding decoder module is modelled using serial and parallel convolutions separately. The residual connections provided in the decoder blocks enable better reconstruction of image parts, leading to remarkable pixelwise prediction. The factorization of more expensive kernel operations into less-expensive ones in the parallel version made the model more highlighted with reduced memory requirement. The dilated convolution, capable of enhancing the receptive field, is utilized in the first residual decoder block that proved to improve the quality of prediction mask further. The encoder and decoder blocks bridged via skip connections in a U shape made the reconstruction highly productive with low-level feature details. With repeated experimental analysis, it has been observed that the model achieved significant capability in separating the semantically relevant objects in the scene.

The model presented is benchmarked on a dataset of road images, which consists of a maximum of 12 classes in a single image. As a future work, the model can be extended to multimodal images, which may contain larger number of classes of objects, of different domains. Also, the model uses a pretrained VGG16 as its encoder, which limits itself to the 13 convolutional layers. The model can be extended with a novel encoder, with state-of-the-art deep learning methodologies, which may improve the segmentation model further.

References

1. O.A. Alzubi, J.A. Alzubi, M. Alweshah, I. Qiqieh, S. Al-Shami, M. Ramachandran, An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Comput. Appl.* 1–17 (2020)
2. P. Sharma, S. Sundaram, M. Sharma, A. Sharma, D. Gupta, Diagnosis of Parkinson’s disease using modified grey wolf optimization. *Cogn. Syst. Res.* **54**, 100–115 (2019)
3. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks. *Commun. ACM* **60**(6), 84–90 (2017)
4. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition (2014). [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
5. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, A. Rabinovich, Going deeper with convolutions, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9
6. D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, K. Dietmayer, Deep multi-modal object detection and semantic segmentation for autonomous driving: datasets, methods, and challenges. *IEEE Trans. Intell. Transport. Syst.* (2020)

7. M. Weigert, U. Schmidt, R. Haase, K. Sugawara, G. Myers, Star-convex polyhedra for 3D object detection and segmentation in microscopy, in *The IEEE Winter Conference on Applications of Computer Vision* (2020), pp. 3666–3673
8. N. Liu, N. Zhang, J. Han, Learning selective self-mutual attention for RGB-D saliency detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 13756–13765
9. C. Li, W. Xia, Y. Yan, B. Luo, J. Tang, Segmenting objects in day and night: edge-conditioned cnn for thermal image semantic segmentation. *IEEE Trans. Neural Netw. Learn. Syst.* (2020)
10. H.Y. Han, Y.C. Chen, P.Y. Hsiao, L.C. Fu, Using channel-wise attention for deep CNN based real-time semantic segmentation with class-aware edge information. *IEEE Trans. Intell. Transport. Syst.* (2020)
11. J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3431–3440
12. H. Liu, Q. Ye, H. Wang, L. Chen, J. Yang, A precise and robust segmentation-based lidar localization system for automated urban driving. *Remote Sens.* **11**(11), 1348 (2019)
13. B. Cheng, L.C. Chen, Y. Wei, Y. Zhu, Z. Huang, J. Xiong, T. Huang, W.-M. Hwu, H. Shi, SPGNet: semantic prediction guidance for scene parsing, in *Proceedings of the IEEE International Conference on Computer Vision* (2019), pp. 5218–5228
14. O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer, Cham, 2015), pp. 234–241, Oct 2015
15. J.A. Alzubi, A. Kumar, O.A. Alzubi, R. Manikandan, Efficient approaches for prediction of brain tumor using machine learning techniques. *Indian J. Public Health Res. Dev.* **10**(2), 267–272 (2019)
16. V. Badrinarayanan, A. Kendall, R. Cipolla, SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017)
17. K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1904–1916 (2015)
18. F.I. Diakogiannis, F. Waldner, P. Caccetta, C. Wu, ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data. *ISPRS J. Photogramm. Remote Sens.* **162**, 94–114 (2020)
19. S. Kohl, B. Romera-Paredes, C. Meyer, J. De Fauw, J.R. Ledsam, K.H. Maier-Hein, O. Ronneberger, A probabilistic U-Net for segmentation of ambiguous images, in *Advances in Neural Information Processing Systems* (2018), pp. 6965–6975
20. N. Ibtehaz, M.S. Rahman, MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural Netw.* **121**, 74–87 (2020)
21. L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(4), 834–848 (2017)

Ensembled Approach for Text Summarization



Minakshi Tomer, Dishant Rathie, and Manoj Kumar

Abstract There is a tremendous amount of data which is present online, and to extract the useful content is a challenging task. The solution is made possible by the introduction of text summarization. In this paper, an ensembled approach for text summarization is proposed, in which the robustness of extractive summarization and abstractive summarization is combined to make the most sense out of the raw data. Extractive text summarization is implemented by using RNN model based on LSTM architecture. The output generated by this model is used as input for the abstractive summarization. Pointer Generator Network is used for implementation of abstractive text summarization. The standard CNN/daily mail dataset is used for experimental purpose. The results are evaluated using ROUGE scores.

Keywords Abstractive text summarization · Extractive text summarization · Recurrent neural network (RNN) · Pointer generator network · Long short term model (LSTM) · Gated Recurrent unit (GRU)

1 Introduction

The process in which a precise, relevant, and instructive summary is generated from large text source document is called text summarization. This process, abruptly, is divided into the following two types: Extractive text summarization and Abstractive text summarization. The process of generating important contextual phrases and words from the given text is incorporated by extractive text summarization,

M. Tomer
USICT, GGSIPU, Delhi, India

Dept IT, MSIT, Delhi, India

D. Rathie (✉)
MSIT, GGSIPU, Delhi, India

M. Kumar
AIACTR, GGSIPU, Delhi, India
e-mail: manoj.kumar@nsut.ac.in

whereas abstractive summary incorporates the process in which the summary generated is synonymous to a human written summary by precisely rephrasing the original source context. In this paper, the Ensembled approach for text summarization which comprises of models based on Extractive and Abstractive summarization is presented.

Ensembled approach for text summarization combines the strength of abstractive and extractive text summarization. The aspect capable of obtaining high ROUGE values with sentence-level attention is the extractive aspect. The only drawback of this aspect is that it is less intellectual. On the other hand, the abstractive aspect is somewhat more complicated than the extractive part and is capable of generating more intellectual summary because of its ability to achieve word-level dynamic attention. In the proposed model, the adjustments done in word-level attention are made by using sentence-level attention, which as a result reduces the chances of generation of words which are present in the sentences which are less attended. By inducing an unconventional inconsistency loss function, the inconsistency between both the levels of attention is castigated. The ROUGE scores have been calculated. The actual losses and the inconsistency losses of both the summarization models along with end-to-end training of the model have been incorporated in calculation of the ROUGE values.

For the extractive summarization, RNN model based on LSTM cell architecture is used and for the abstractive summarization, a hybrid Pointer Generator network is used.

The recurrent neural network (RNN) architecture capable of remembering the values over random intervals is LSTM. This is the same reason it is widely used for categorizing, processing, and anticipating time series over given time lags whose duration is unknown.

Pointer Generator Network is the only sequence-to-sequence model which is capable of production of such an output text sequence in which the elements that are a part of input text sequence are also used.

The organization of other parts of paper is as follows: Sect. 2 contains the literature survey of the text summarization. Section 3 describes the proposed model with various techniques utilized in it. Section 4 consists of the experiment performed and results obtained, which is followed by the conclusion of the paper along with future scope in Sect. 5.

2 Literature Survey

The process of abbreviating text information into a precise edition is called Text Summarization [1]. The process is conducted automatically and important points are maintained throughout. This ability is helpful in aiding many applications such as generation of reports, creation of news digests, and presentation of search results. The approaches followed in text summarization are widely classified in the following two types: abstractive text summarization and extractive text summarization. The approaches in which, the summary generated is directly combined from the source

text by selecting complete sentences at any instance of time, are the extractive approaches. On the contrary, the approaches where novel words and phrases are generated, which are non-existent in the original text, are abstractive approaches. Therefore, the abstractive approaches are more coherent and intellectual in comparison to the extractive approaches. The extractive approaches are relatively simpler. Furthermore, the probability of the selection of any sentence into the summary is obtained as output.

A novel approach for abstractive text summarization has been evident from the Neural sequence-to-sequence models [2]. However, these models tend to reproduce factual details inappropriately, and they are also liable to repetition.

In the summarization process, the keywords which are at the core of the summary are often omitted. This is so because they are very few in number with respect to the data used for training the model. The decoder also cannot decode those words because its vocabulary is fixed at the time of training, which makes it almost impossible for it to take those words into consideration [3]. An instinctive way to solve the neglection of such OOV words is by pointing their location in the parent document, which is made possible by the proposed pointer generator architecture.

In the various Natural language processing tasks such as question answering and dialog delivery, systems generate text from a structured data [4]. The most prominent frameworks used for table-to-text generation are encoder-decoder or neural language models.

There is an abundant amount of data available to us due to the increasing emergence of availability of the Internet. The summarization and analysis of this huge amount of raw and unprocessed data is very difficult for a human brain. So, with this data flow, the requirement of tools to summarize this data is of utmost importance. In the paper [5], emphasis was given on single and multi-document summarization using extractive approaches.

The performance of encoder decoder models, based on the RNN architecture, were at par with state-of-the-art models for the abstractive approaches even if the sequences of input and output were kept short. But when the scope of these models was expanded and were applied on texts of larger size, the output generated consisted of repetitive and incoherent phrases. In the paper [6], a new training method was introduced which induces a neural network model with intra-attention. In this, methodology reinforcement learning (RL) and a standardized word prediction model was combined.

A hybrid pointer generator network [2] (with a bidirectional GRU Encoder and a bidirectional LSTM Encoder) can copy the words from the parent or the source text via pointing and secondly the coverage mechanism keeps a track of the context which is already summarized.

The reviews of the approaches based on deep learning [7] are hopeful in the search for solving the abstractive text summarization which was an unsolved mystery up until now and they are quite promising too. But the multi-sentence summarization is yet to be generalized because of the lack of data for training and the metric.

Generation of inappropriate and insufficient content is still an issue for most of the preexisting models in the abstractive approaches. In [8], a method to generate

summary, which is true to its topic and aware of its context, is suggested. To do this, the intellectual quality and effectiveness of the summary is upgraded by inducing keywords from the source text by pointing towards them using a pointer generator network.

3 Proposed Model

The Proposed model combines the pros and eliminates the cons of both broad techniques used in text summarization by connecting extractor (modified Nallapati et al. 2017) and abstractor. Sentence-level as well as word-level attentions are combined in the proposed model (Fig. 1).

3.1 Combining Attentions

As stated in (Vaswani et al. 2017), attention mechanism is a key factor in the tasks of Natural Language Processing. Thus, by using simple scalar multiplication and renormalization the explicit combination of sentence-level and word-level attentions are introduced. The updated word attention is

$$\hat{\alpha}_m^t = \frac{\alpha_m^t \beta_{n(m)}}{\sum_m \alpha_m^t \times \beta_{n(m)}} \quad (1)$$

The values of both sentence and word-level attentions are high, which is ensured by multiplication, so that updated word attention is also high. Extractor has already achieved high ROUGE value which ensures good sentence-level attention. Also, the spurious word-level attention is mitigated by the sentence-level attention as

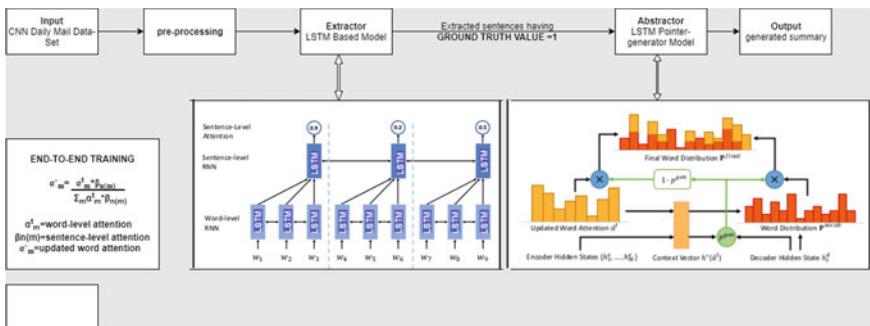


Fig. 1 Ensembled approach for text summarization

words that are present in the sentences which are less visited have low probability of generation in the summary. Thus, extractor supports abstractor to improve its performance.

3.2 Extractor

The proposed extractor is a little bit modified from (Nallapati et al. 2017). Major modification is that the final summary is not obtained by extractor. Functionality of extractor is to obtain lesser number of more important sentences with high recall, which can further facilitates better function of the abstractor.

The performance achieved in the tasks of processing sequences of variable length using RNN-based approach is the state of the art. Moreover, the training of RNN model consisting of gated units is easy as compared to the vanilla RNN and produce better results in various tasks.

Therefore, LSTM-based RNN is used to improve its performance of vanilla RNN. A LSTM can be defined as the following equations:

$$\begin{aligned} f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\ i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\ o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\ h_t &= o_t \circ \sigma_h(c_t) \end{aligned} \quad (2)$$

where

x_t : input vector of LSTM unit.

f_t : forget gate's activation vector.

i_t : input gate's activation vector.

o_t : output gate' activation vector.

h_t : vector of hidden state.

c_t : vector of cell state.

Hierarchical bidirectional LSTM predicts sentence-level attention for each sentence by extracting sentence representations and classification layer.

The following sigmoid cross entropy loss is used

$$L_{ext} = -\frac{1}{N} \sum_{n=1}^N (g_n \log(\beta_n) + (1 - g_n) \log(1 - \beta_n)) \quad (3)$$

where the ground-truth label for nth sentence is represented by $g_n = (0,1)$ and the number of sentences is represented by N.

The indication of the nth sentence to be attended and facilitate the abstractive summarization is shown when $g_n = 1$.

Ground-truth label: To extract sentences with high informativity is the main goal of extractor, this means that the amount of information contained in the extracted sentences, which is further used to generate abstractive summary, must be as much as possible.

3.3 Abstractor

Pointer generator network is the second part of the proposed model. This approach consists of two parts: first pointing and then generating. The words which are important to the summary are pointed in the source document (via copying) and later the words from a fixed vocabulary are generated. Equation 3 depicts the calculation of a: attention distribution and h: context vector. h_t : context vector, s_t : decoder state and x : decoder input is used for evaluation of probability of generation p ($0 \leq p \leq 1$).

$$p_{gen} = \sigma(w_h^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr})_\infty \quad (4)$$

Learnable parameters are represented by w_h^t , w_s^t , w_x^t , and b_{ptr} . σ represents sigmoid function. p_{gen} : parameter used to decide where the word will be taken from. Whether it will be taken from the source document or the word will be generated from p_{vocab} .

The extended vocabulary for each document denotes the union of all words in the vocabulary and words appearing in the source document (4)

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w_i=w} a_i^t \quad (5)$$

$p_{vocab}(w)$ is zero when a case arises where w is an OOV. In the same manner, $\sum_{i:w_i=w} a_i^t = 0$ when there is no such word present in the document. The ability of the pointer generator model which is highlighted here is its ability in the production OOV words.

3.4 Coverage Mechanism

Prior to the introduction pointer generator network, the problem of repetition was very prominent in the model. The proposed model is an adaptation of the coverage model applied in [2]. The coverage vector retained in the model is represented by

c. It is the sum of all the attentions distributed all over the decoder in previous time steps.

Intuitively, the degree of coverage that has been received by words from the attention mechanism is represented by c .

An extra input in the form of coverage vector is given to the attention mechanism and is represented as (5):

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + w_c c_i^t + b_{attn}) \quad (6)$$

In order to avert the redundancy of the generated text, a learnable parameter w_c is introduced. The length of this vector is same as v . This ensures that the attention mechanism is making decisions by keeping in mind the preceding decisions it took (5).

To penalize the redundancy, a coverage loss is described additionally (6):

$$\text{covloss}_t = \sum_i \min(a_i^t, c_i^t) \quad (7)$$

As the requirement of summarization does not necessarily involve uniform coverage, the nature of loss function is robust. This allows us to only penalize the existing overlap between the coverage and each attention distribution. Thus, preventing redundancy.

The formation of basic pointer generator network is carried out by the two encoders: bidirectional LSTM encoder and bidirectional GRU encoder along with a unidirectional LSTM.

4 Experiment and Results

4.1 Dataset

Dataset used is CNN/Daily Mail dataset [3, 9]. It is a standardized dataset. It consists of news articles of token size 781 as an average, along with their corresponding reference summary with an average token size of 56. The code used is provided by Nallapati et al. [3]. The data which is obtained has train size of 287,225, validation size of 13,360, and test size of 11,480.

4.2 Implementation:

The size of vocabulary used by the designed pointer generator network is 50k words. This size was found sufficient to handle OOV words. All the models used for experimentation consist of 128-dimensional word embeddings and 256-dimensional hidden state.

The word embeddings were learned from scratch during the training of the model using [10] and not pretrained as suggested in [3]. The initial accumulator value was kept at 0.1 and the learning was kept at 0.15.

Training each model took 32 h approximately. The models trained for about 35,000 epochs with batch size of 16.

The article was trimmed down to 500 tokens at the time of training in order to improve the training and the testing phase and there was limitation set on the size of summary to 120 tokens while training and 150 while testing. This resulted in the improvement of the efficiency of the entire model at the time of training.

Beam search was used to generate the summaries at the time of testing. The size of beam in beam search was set to 4. The improvement in the efficiency and the utilization of coverage mechanism to its maximum value was done by the help of highly trimmed sequence.

4.3 Training Procedure

End-to-end Training. In this sentence level attention (β soft attention) and word level attention (α_t) are used. Loss function is minimized, the batch size is kept at 8 due to the availability of the memory. For this batch size the learning rate is kept at 0.1.

4.4 Result

See Table 1 and Fig. 2.

Table 1 The below table depicts the ROUGE values (based on three basic parameters, namely: fscore, recall, and precision) of the models

ROUGE values	Ensembled approach comprising of LSTM cell	Ensembled approach comprising of GRU Cell	Pointer generator network based on Bi-directional LSTM	Pointer generator network based on Bi-directional GRU
Rouge_1_f_score	0.3608	0.3558	0.3514	0.2505
Rouge_1_recall	0.3788	0.3756	0.3664	0.2357
Rouge_1_precision	0.3672	0.3660	0.3260	0.2937
Rouge_2_f_score	0.1658	0.1572	0.1553	0.0552
Rouge_2_recall	0.1702	0.1724	0.1629	0.0530
Rouge_2_precision	0.1625	0.1603	0.1575	0.0607
Rouge_L_f_score	0.3418	0.3334	0.3271	0.2320
Rouge_L_recall	0.3538	0.3434	0.3408	0.2184
Rouge_L_precision	0.3403	0.3357	0.3316	0.2725

5 Conclusion and Future Scope

On conducting a comparative analysis, it has been concluded that Ensembled approach for text summarization comprising of RNN-based extraction based on LSTM cell architecture has shown better ROUGE Scores as compared to the Ensembled approach for text summarization comprising of RNN-based extraction based on GRU cell architecture.

Since, ensembled approach for text summarization is the most recent model, so any of the models before it is not suggested. And further, to improve the results of the proposed model, implementation of different optimization can be done and the study of those results can be conducted thereafter.

Original article:

Never mind cats having nine lives .A stray pooch in washington state has used up at least three of her own after being hit by a car , apparently whacked on the head with a hammer in a misguided mercy killing and then buried in a field only to survive .that's according to washington state university , where the dog -- a friendly white-and-black bully breed mix now named theia -- has been receiving care at the veterinary teaching hospital .four days after her apparent death , the dog managed to stagger to a nearby farm , dirt-covered and emaciated , where she was found by a worker who took her to a vet for help .she was taken in by moses lake , washington , resident sara mellado .considering everything that she's been through , she 's incredibly gentle and loving , mellado said , according to wsu news .she's a true miracle dog and she deserves a good life . 'theia is only one year old but the dog 's brush with death did not leave her unscathed .the veterinary hospital 's good samaritan fund committee awarded some money to help pay for the dog 's treatment , but mellado has set up a fundraising page to help meet the remaining cost of the dog 's care .she 's also created a facebook page to keep supporters updated .donors have already surpassed the \$ 10,000 target , inspired by theia 's tale of survival against the odds .on the fundraising page , mellado writes , `` she is in desperate need of extensive medical procedures to fix her nasal damage and reset her jaw .i agreed to foster her until she finally found a loving home .she is dedicated to making sure theia gets the medical attention she needs , mellado adds , and wants to `` make sure she gets placed in a family where this will never happen to her again ! any additional funds raised will be `` paid forward " to help other animals .theia is not the only animal to apparently rise from the grave in recent weeks .a cat in tampa , florida , found seemingly dead after he was hit by a car in january , showed up alive in a neighbor's yard five days after he was buried by his owner .the cat was in bad shape , with maggots covering open wounds on his body and a ruined left eye , but remarkably survived with the help of treatment from the humane society .

Reference:

theia , a bully breed mix , was apparently hit by a car , whacked with a hammer and buried in a field .she's a true miracle dog and she deserves a good life , " says sara mellado , who is looking for a home for theia .

Ensembled approach for text summarization:

stray pooch in washington state has used up at least three of her own after being hit by a car , apparently whacked on the head with a hammer in a misguided mercy killing and then buried in a field .four days after her apparent death , the dog managed to stagger to a nearby farm , dirt-covered and emaciated , where she was found by a worker who took her to a vet for help .

Fig. 2 Comparison between reference output and ensembled approach for text summarization generated output

References

1. W.-T. Hsu, C.-K. Lin, M.-Y. Lee, K. Min, J. Tang, M. Sun, A unified model for extractive and abstractive summarization using inconsistency los. [arXiv:1805.06266](https://arxiv.org/abs/1805.06266) [cs.CL] July 2018
2. A. See, P. Liu, C. Manning, Get to the point: summarization with pointer-generator networks, in *Association for Computational Linguistics*, [arXiv:1704.04368v2](https://arxiv.org/abs/1704.04368v2), [cs.CL] Apr. 2017.
3. R. Nallapati, B. Zhou, C. dos Santos, C. Gulcehre, B. Xiang, Abstractive text summarization using sequence-to-sequence RNNs and beyond. [arXiv:1602.06023v5](https://arxiv.org/abs/1602.06023v5) [cs.CL] 26 Aug 2016
4. L. Sha, L. Mou, T. Liu, P. Poupart, S. Li, B. Chang , Z. Sui. “Order-Planning Neural Text Generation From Structured Data.” arXiv preprint [arXiv:1709.00155](https://arxiv.org/abs/1709.00155) (2017)
5. M. Allahyari, et al., Text summarization techniques: a brief survey. Arxiv Preprint [arXiv:1707.02268](https://arxiv.org/abs/1707.02268) (2017)
6. R. Paulus, C. Xiong, R. Socher, A deep reinforced model for abstractive summarization (2017)
7. P. Devihosur, et al., Automatic text summarization using natural language processing (2017)
8. X. Jiang, P. Hu, L. Hou, X. Wang, Improving pointer-generator network with keywords information for chinese abstractive summarization, in *7th CCF International Conference, NLPCC 2018*, Hohhot, China, August 26–30, 2018, Proceedings, Part I. (pp. 464–474, 2018). https://doi.org/10.1007/978-3-319-99495-6_39
9. K.M. Hermann, T. Kociský, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, P. Blunsom, Teaching machines to read and comprehend, in *Neural Information Processing Systems* (2015)
10. J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.* **12**, 2121–2159 (2011)

Statistical Analysis of Impact of COVID-19 Pandemic on States of India



Prerna Pandey, Nikki Saraswat, Priyansh Shukla, Kavita Sharma, and Shiv Naresh Shivhare

Abstract Novel coronavirus, i.e., COVID-19 affected the human lifestyle to a great extent worldwide. Economy, health, and employment are among the severely affected sectors of India, which suffered in many aspects. In this paper, we propose an efficient algorithm to calculate and analyze the impact of COVID-19 in several states of India, through which the recovery process can be planned and initialized. The proposed method analyzes the overall impact of coronavirus by considering several significant attributes from different sectors, viz., death rate, COVID-positive cases, testing, GDP rate, inflation rate, etc. The experiments were conducted on the real and authentic dataset obtained from Indian govt. websites. Each state is assigned a grade based on the score calculated using statistical analysis which represents the impact of COVID-19. The proposed method presents a hypothesis which can be really useful for planning and implementing the rehabilitation and recovery from COVID-19 in India.

Keywords COVID-19 · Coronavirus · Statistical Analysis

1 Introduction

The world is now facing a huge crisis due to the novel coronavirus (COVID-19). In India, there were 36,19,174 confirmed cases and 65,435 deaths recorded as of 31st August 2020. The consequence of this lockdown was the closure of businesses and non-essential services, educational institutions, holy places, production plants, transportation of goods, inter and intra-state traveling, mass gatherings, malls, cinema halls, trains, and metro, etc., across all states and union territories of India. Due to this, India suffered greatly in three major sectors, i.e., the health sector, economic

P. Pandey (✉) · N. Saraswat · P. Shukla · S. N. Shivhare
School of Computer Science, University of Petroleum and Energy Studies, Dehradun,
Uttarakhand, India

K. Sharma
Department of CSE, G. L. Bajaj Institute of Technology and Management, Greater Noida, India

sector, and employment sector. On 11th March 2020, World Health Organization (WHO) proclaimed COVID-19 as a global pandemic. In December 2019, COVID-19 was first detected in Wuhan, China, after which it has affected almost 26 million people across the globe. On 30th January 2020, COVID-19 was first found in Kerala, India. The government of India had announced lockdown in different phases. The coronavirus has led to a dramatic loss of human life and challenged public health the most.

Coronavirus had been a substantial shock for the world economy, as well. Only China has shown positive growth in the world. The Indian economy has been hit hard by the pandemic and unprecedented lockdown that continued for almost 4 months. The Indian economy was already in a dreadful state and was recovering from demonetization and early GST rollout. Coronavirus has thus turned out to be the last nail in the coffin. The downfall will continue in the fiscal year 2020–2021 as well. International Monetary Fund (IMF) predicted a downfall of 23.9% this quarter owing to the current situation due to coronavirus. The downfall of the economy will affect employment along with the per capita income. GDP growth for the fiscal year 2019–20 was the lowest since 2002–03 and was set at a 4% growth rate. The informal sector, banking sector, corporate sector, and agriculture sector have also suffered a loss due to the downfall of the economy. It is predicted that it will take almost a year for the Indian economy to reach the pre-COVID-19 state with recovery starting as early as the lockdown is lifted. It will also hugely depend on government schemes and RBI policies that can play a crucial role in the upbringing of this downfall. Due to the nationwide lockdown caused by COVID-19, there has been a huge increase in the unemployment rate not just in India, but across the Globe. A large portion of the working population lost their employment due to this pandemic. This is an unprecedented recession, unlike any in the past.

Another major fallback that India faced during COVID-19 was the migration of workers from their workplace to their native place. The migrant workers were left unemployed, due to the shutdown of industries, power plants, construction sites, factories, etc. However, given the regional diversity of India, analysis of the impact of coronavirus by observing the positive cases across the country, i.e., the state-wise analysis seems feasible. An accurate estimate and analysis are needed to measure the impact of COVID-19 in Indian society and further required to plan and initiate the efficient recovery process. In this paper, we showcased the in-depth analysis of COVID-19 concerning different sectors of different states of India, with the help of mathematical analysis of data. The sectors that were taken into account were health, employment, and the economy of Indian states. These sectors represent the major areas of states which control their overall economy as well as the well-being of their people into different categories (grades). Furthermore, an appropriate weightage for each attribute depending on its effect on the sector and its relevance is considered.

The major contribution of this paper is two-fold:

1. An efficient algorithm is proposed and implemented to estimate the state-wise impact of COVID-19 in India, which might be helpful to plan the recovery planning and further actions.

2. The classification of the state of India is performed by considering the impact of the pandemic on major sectors, i.e., the health sector, economic sector, and employment sector.

The remainder of the paper is structured as follows: Section 2 presents a survey of numerous recent and significant methods of the domain. Section 3 discusses the proposed method by providing the details of each step. Section 3 demonstrates the experimental results of the proposed method in terms of significant graph modeling and state-wise grading. Finally, concluding remarks are given as per the conducted experiments and analysis in Sect. 4.

2 Related Work

In a recent study [1], it was speculated that the number of coronavirus positive cases was increasing very rapidly in India and due to the absence of any vaccine or therapy, there is no way to end this worldwide pandemic. A mathematical model was prepared that predicts the dynamics of COVID-19 with its end date in 17 provinces of India. The model monitors the six major compartments of individuals who are susceptible, asymptomatic, recovered, infected, isolated infected, and quarantined susceptible individuals. It was predicted that 95% population of India will be infected by 26 June 2020 and 99% by 27 July 2020 which was also predicted as the end date of COVID-19 in India. In [2], the symptoms and physical effects of coronavirus are analyzed and a statistical study on the number of patients around the world is conducted. The authors concluded that the spread of this virus is independent of the age of a person, but gender and cause of transmission have a significant relationship with it. Moreover, Sengupta et al. [3] presented the analysis of the effects of coronavirus on various sectors such as tourism, capital markets, petroleum, retail market, aviation, real estate, hospitality, etc. The social and financial effect is monitored through the work hours with an examination of the formative approach.

Xiong et al. [4] showcased that COVID-19 has not only affected the economy and employment, but also the mental health of the people. An analysis is conducted to measure the effect of COVID-19 on the mental health of the population and the risks related to mental health. In the end, it was concluded that symptoms of anxiety, depression, post-traumatic distress disorder, and stress were diagnosed among the population of China, Italy, Spain, Iran, and Nepal during the COVID-19 pandemic period. Also, risks related to health were reported in the young generation due to unemployment, for which clinical support is required. Another study was conducted on the citizens of Saudi Arabia to find the impact of the pandemic on their mental health [5]. The results of the study were clearly showing that during the early stage, one-fourth of the population was experiencing moderate effects on mental health, but some specific precautionary measures appeared to have positive effects on the mental health of the people. Furthermore, Chatterjee et al. [6] presented a stochastic mathematical model to study the effect of COVID-19 on healthcare. It was predicted

in April 2020, that with the growth of the pandemic, India's healthcare resources will be finished by the end of May 2020, but with the help of institutions like National Provider Identifier (NPI), the total number of confirmed cases, hospitalizations, deaths, and ICU requirements can be reduced by almost 90%. COVID-19 pandemic has also affected the suicide rate. According to another study conducted by Sher et al. [7], it was observed that the number of suicide cases has increased during and after the COVID-19 pandemic. The reasons for suicide are also recorded with the major symptoms. In [8], a Geo-social gradient with predicted diseases has been found across Great Britain by collecting data from a study app and it was observed that urban areas and areas with higher deprivation are most affected and also the results help the citizens to find the major symptoms in that particular geographical area so that precautions can be taken to avoid the disease.

3 Proposed Method

We have conducted the proposed analysis into three major and significant sectors of India, such as the health sector, the employment sector, and the economic sector. Effect of COVID-19 on the selected sector is computed based on several attributes, i.e., Health sector: confirmed cases, death cases, tested cases and recovery rate, Employment sector: unemployment rate and labour participation rate, Economy sector: GDP (Gross Domestic Product), inflation rate and growth percentage of GST. For each attribute, a weight is assigned based on its importance and contribution and the total score of that attribute is calculated using the following mathematical formulation (Fig. 1):

Figure 2 displays the flowchart of our project. It starts with collection of raw data for the project. Data collection for all individual sector is the first step of research analysis. This step also includes cleaning of data and storing it in a file to be used for further processing. After this, pick a sector and calculate the score of each attribute

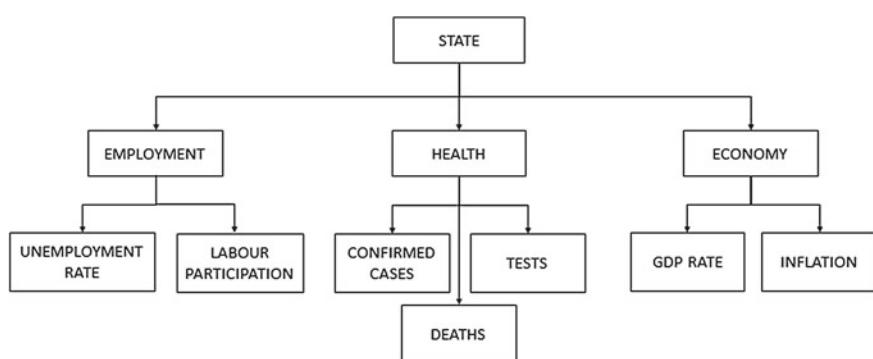


Fig. 1 Hierarchy chart of attributes for different sector

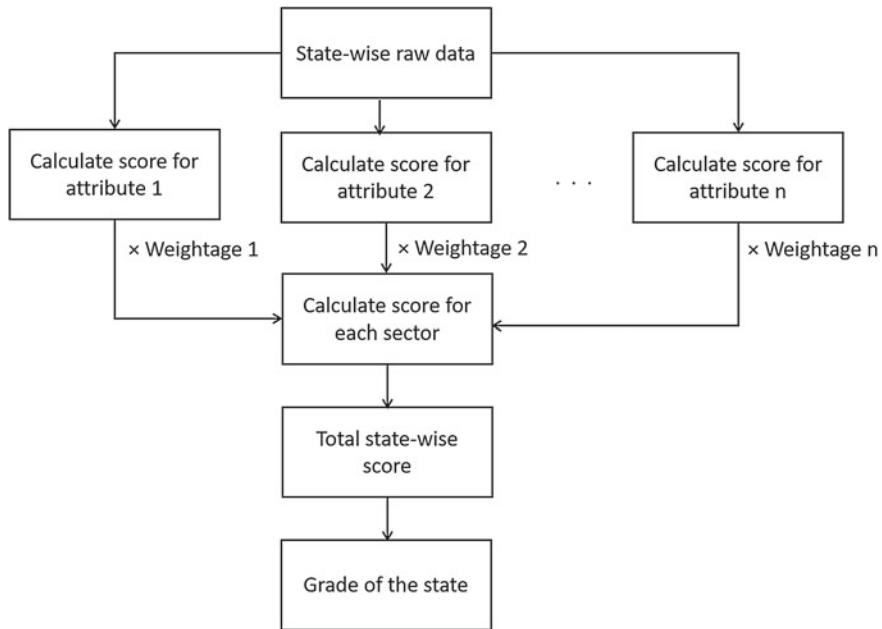


Fig. 2 Flowchart of the proposed method

in it. The weightage multiplication is done in each attribute based on the importance of that attribute of that particular sector, for example, death in health is the most important factor, and thus will be assigned a higher weightage as compared to its fellow attributes. Summation of all the attributes score gives the final score of that particular sector. After we have the final scores of all the three sectors, a final score is calculated based on which each state is divided into different grade category. This completes the last stage of the following flowchart.

The overall step-wise procedure of the proposed method to analyze the impact of COVID-19 in different states of India can be described in the form of the following algorithm:

Health Sector: Weightage of Deaths, $W_1 = 1.0$, Weightage of Tests, $W_2 = 0.10$, Weightage of Confirmed Cases, $W_3 = 0.75$. Weightage of death is assigned the maximum, i.e., 1, since health is the most important factor affecting the health of a particular state. The same goes for weightage of other attributes. Equation for calculating scores for each attribute

$$\text{Death Score} = W_1 \times \frac{N_1}{P} \quad (1)$$

$$\text{Tested Score} = W_2 \times \frac{N_2}{P} \quad (2)$$

$$\text{Confirmed Cases Score} = W3 \times \frac{N_3}{P} \quad (3)$$

where, $N1$ = Number of deaths, $N2$ = Number of Tests, $N3$ = Number of Cases, and P = Population of the state.

Employment Sector: Weightage of Unemployment Rate, $W4 = 1.0$, Weightage of Labour Participation Rate, $W5 = 0.5$. Weightage of unemployment rate is maximum in employment sector. Unemployment rate is comparatively more important than labour participation rate for any state.

Equation for calculating scores for each attribute:

$$\text{Unemployment Score} = W4 \times (U2020 - U2019) \quad (4)$$

$$\text{Labour Participation Score} = W5 \times (L2020 - L2019) \quad (5)$$

where $U2020$ = Average unemployment of 2020 (from March 2020 to August 2020), $U2019$ = Average unemployment of 2019 (from March 2019 to August 2019), $L2020$ = Labour participation Rate 2020, and $L2019$ = Labour participation Rate 2019.

Economic Sector: Weightage of GDP, $W6 = 1.0$, Weightage of Inflation Rate, $W7 = 0.5$. Economic sector in itself is one of the most important sectors for any country. GDP is the prime factor that is seen as the most important comparison tool among countries. For the same, GDP is assigned a maximum weightage of 1 followed by weightage of inflation rate. Equation for calculating scores for each attribute:

$$\text{GDP Score} = W6 \times (G2020 - G2019) \quad (6)$$

$$\text{Inflation Score} = W7 \times I \quad (7)$$

where $G2020$ = GDP of 2020, $G2019$ = GDP of 2019, and I = Rate of Inflation for 2020.

Attributes scores are normalized to bring scores in a given range for the purpose of comparing and evaluating total score. After evaluating the score of each attribute in all the above sectors, final score of each sector is calculated, which is termed as health score, employment score, and economy score.

$$\text{Health Score} = \text{Tested Score} - \text{Death Score} - \text{Confirmed Cases Score} \quad (8)$$

$$\text{Employment Score} = \text{Labor Participation Score} - \text{Unemployment Score} \quad (9)$$

$$\text{Economy Score} = \text{GDP Score} - \text{Inflation Score} \quad (10)$$

Table 1 Grades assigned to states and their description

Grade	Description
O	Well performing state
A, B, C	Average performing state
D	Poor performing state

Final score for each State is calculated by using the given formula:

$$\text{Final Score} = \text{Health Score} + \text{Employment Score} + \text{Economy Score} \quad (11)$$

According to the final score, the states are categorized into different categories and each category has a different grade as follows (Table 1).

4 Experimental Results

Furthermore, Fig. 3 shows the result of health sector in the form of a bar graph. It can be visualized that Maharashtra and Puducherry are the poor and well performing.

Figure 4 is used to represent the *EmploymentScore* of each state mentioned in Table 2 in the form of a bar graph for better visualization. It can be visualized that Puducherry and Sikkim are the poor and well-performing states. Figure 5 represents the result as bar graph. It can be visualized that Maharashtra and Assam are the well and poor performing in the economic sector. Final score for each State is calculated by using the Eq. (11):

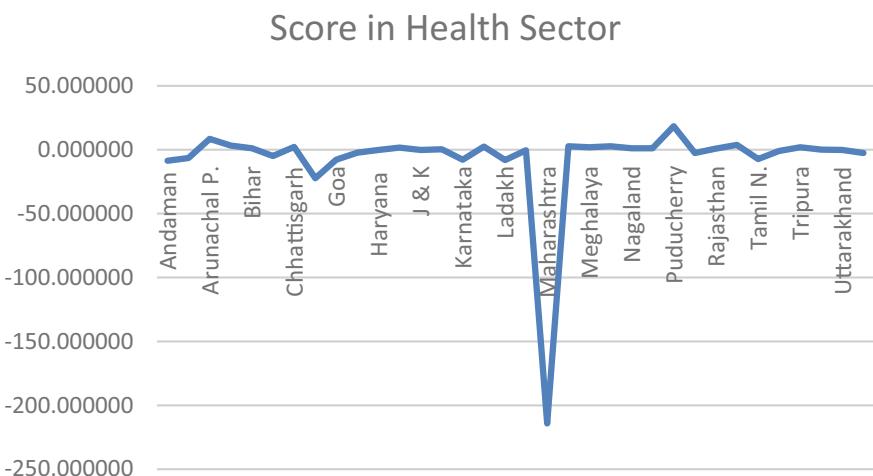


Fig. 3 Bar Graph for health score

Score in Employment Sector

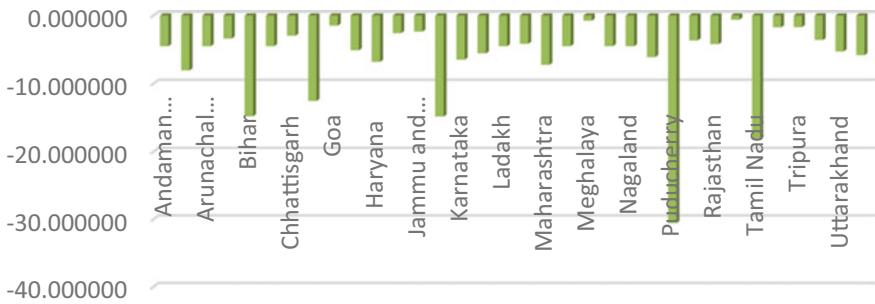


Fig. 4 Bar graph for employment score

Score for Economy Sector

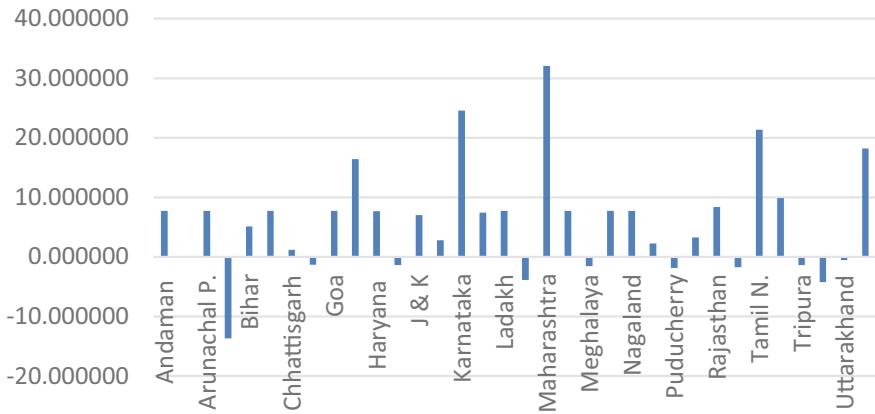


Fig. 5 Bar graph for economic sector

The score of each sector is calculated using Eqs. (8), (9), and (10) given above. The scores for each sector are given in Table 2.

4.1 Discussion

Coronavirus came as an unprecedented shock to the World. Every country in the world suffered from its unfortunate effects. In India, there were 36,19,174 confirmed cases and 65,435 deaths as of 31st August 2020. The Indian Economy was already crumbling before COVID-19 hit us. Months-long nationwide lockdown has worsened

Table 2 Scores for Health, Employment, and Economy Sector for each state

Sr	State	Health	Emp	Economy	Final score	Grade
1	Andaman	-8.62	-4.51	7.71	-5.42	A
2	Andhra Pradesh	-6.51	-8.06	0.10	-14.47	B
3	Arunachal Pradesh	8.41	-4.51	7.71	11.61	O
4	Assam	3.19	-3.37	-13.69	-13.88	B
5	Bihar	1.18	-14.81	5.07	-8.56	A
6	Chandigarh	-5.02	-4.51	7.71	-1.82	A
7	Chhattisgarh	2.26	-2.98	1.20	0.48	A
8	Delhi	-22.31	-12.56	-1.30	-36.17	C
9	Goa	-7.73	-1.49	7.71	-1.50	A
10	Gujarat	-2.21	-5.12	16.38	9.04	A
11	Haryana	-0.08	-6.81	7.64	0.73	A
12	Himachal Pradesh	1.70	-2.59	-1.37	-2.27	A
13	Jammu and Kashmir	-0.11	-2.37	6.98	4.49	A
14	Jharkhand	0.46	-14.85	2.75	-11.62	B
15	Karnataka	-7.94	-6.48	24.57	10.13	A
16	Kerala	2.32	-5.57	7.41	4.17	A
17	Ladakh	-7.99	-4.51	7.71	-4.79	A
18	Madhya Pradesh	-0.57	-4.15	-3.89	-8.63	A
19	Maharashtra	-214.19	-7.25	32.05	189.39	D
20	Manipur	2.75	-4.51	7.71	5.95	A
21	Meghalaya	1.81	-0.76	-1.53	-0.48	A
22	Mizoram	2.68	-4.51	7.71	5.88	A
23	Nagaland	1.03	-4.51	7.71	4.23	A
24	Odisha	1.01	-6.13	2.25	-2.871	A
25	Puducherry	18.39	-30.58	-1.85	-50.83	C
26	Punjab	-2.64	-3.66	3.29	-3.02	A
27	Rajasthan	0.791	-4.21	8.39	4.96	A
28	Sikkim	3.75	-0.62	-1.74	1.38	A
29	Tamil Nadu	-7.35	-18.13	21.36	-4.11	A
30	Telangana	-1.00	-1.75	9.84	7.08	A
31	Tripura	1.91	-1.72	-1.40	-1.21	A
32	Uttar Pradesh	0.11	-3.60	-4.24	-7.72	A
33	Uttarakhand	-0.24	-5.28	-0.56	-6.09	A
34	West Bengal	-2.56	-5.82	18.19	9.80	A

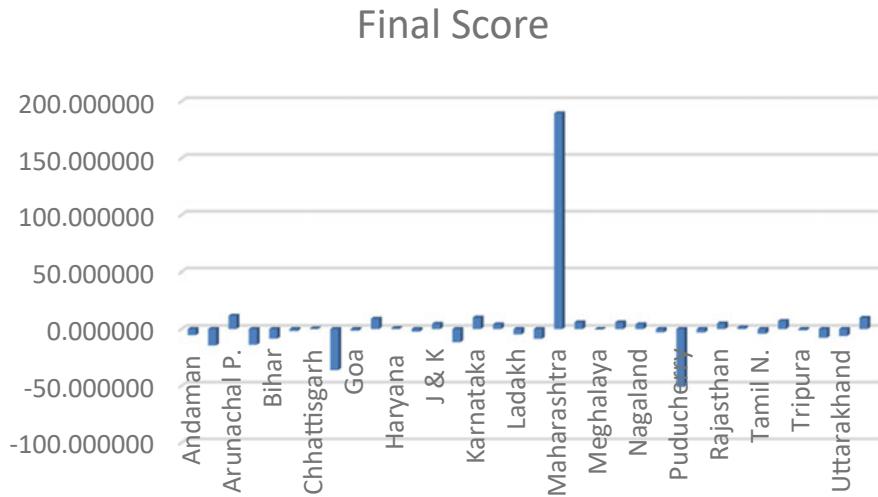


Fig. 6 Bar graph for final score

the situation. As of now, there is no vaccine for this disease, social distancing and a nationwide lockdown was the only solution to keep up with this pandemic. In this research, we analyze the effect of COVID-19 on different sectors concerning 34 States and Union territories of India. The impact of COVID-19 is analyzed on various sectors such as health, economic, and employment sectors. The different parameters used for the analysis are death rate, COVID-positive cases, testing, GDP rate, inflation rate, etc. The state-wise analysis is performed based on the real dataset obtained from authentic Government portals. The impact of the pandemic observed in different states of India is analyzed using statistical analysis through a marking scheme. For each state, scores are calculated based on each sector. These scores are computed to draw a final score, based on which a grade is assigned to each state. The grade demonstrates the degree of effect of COVID-19 on the state.

As compared to the other research papers, the analysis performed is state-wise, including the union territories of India. Also, the sectors included are the most significant sectors on the basis of which the effects of COVID-19 on a state can be measured. In some papers, the parameters considered for the prediction were based on the individuals and are very different from the parameters considered in this analysis. Most of the papers are based on one of these sectors: health, employment or economy, whereas in this analysis, all three sectors are included together. Each state is analyzed closely in each aspect and the scores obtained are used for comparison among the states.

5 Conclusion

From the results, it can be concluded that in the Health Sector, Maharashtra has the lowest score, due to high number of confirmed cases and deaths. Whereas, Puducherry has the best score in this sector. In Employment Sector, Puducherry and Sikkim had the worst and the best grades, respectively. Due to increase in unemployment after COVID-19, Puducherry had the minimum score. Further, Maharashtra has done the best in Economy Sector and Assam has the lowest economy score. After adding the scores of each sector, the final scores were calculated. From the final scores, grades were assigned to each state. Maharashtra had the worst grade, D, and Arunachal Pradesh had the best grade, O. Subsequently, it can be concluded that the worst impact of COVID-19 was on Maharashtra. This paper will help the government bodies in making state-wise recovery plans and utilizing the available resources optimally. This will also help people and the state government to map out the most affected sector in a particular state.

References

1. K. Sarkar, S. Khajanchi, J. Nieto, Modeling and forecasting the COVID-19 pandemic in India. *Chaos, Solitons & Fractal* **139**, 110049 (2020)
2. V. Bhatnagar, R.C. Poonia, P. Nagar, S. Kumar, V. Singh, L. Raja, P. Dass, Descriptive analysis of COVID-19 patients in the context of India. *J. Interdiscip. Math.*, 1–16 (2020)
3. S.M. Dev, R. Sengupta, Covid-19: impact on the Indian economy, in *Indira Gandhi Institute of Development Research, Mumbai April* (2020)
4. J. Xiong, O. Lipsitz, F. Nasri, L.M. Lui, H. Gill, L. Phan, ..., R.S. McIntyre, Impact of COVID-19 pandemic on mental health in the general population: a systematic review. *J. Affect. Disord.* (2020)
5. A.A. Alkhamees, S.A. Alrashed, A.A. Alzunaydi, A.S. Almohimeed, M.S. Aljohani, The psychological impact of COVID-19 pandemic on the general population of Saudi Arabia. *Compr. Psychiatry* **102**, 152192 (2020)
6. K. Chatterjee, K. Chatterjee, A. Kumar, S. Shankar, Healthcare impact of COVID-19 epidemic in India: A stochastic mathematical model. *Med. J. Armed Forces India* (2020)
7. L. Sher, The impact of the COVID-19 pandemic on suicide rates. *QJM: Int. J. Med.* **113**(10), 707–712 (2020)
8. R.C. Bowyer, T. Varsavsky, E.J. Thompson, C.H. Sudre, B. A. Murray, M.B. Freidin, ..., M.J. Cardoso, Geo-social gradients in predicted COVID-19 prevalence in Great Britain: results from 1960242 users of the COVID-19 Symptoms Study app. *Thorax* (2020)
9. Admin, Group of volunteers. Covid19 India. <https://www.covid19india.org/>. Last accessed: Aug 2020.
10. Admin, Centre for monitoring Indian economy. Unemployment Rate in India. <https://unemploymentinindia.cmie.com/>. Last accessed: Aug 2020.
11. Admin, Government of India. Unique Identification Authority of India. <https://uidai.gov.in/>. Last accessed: Sep 2020.
12. Admin, Government of India.GST Council of India. <http://gstcouncil.gov.in/>. Last accessed: Sep 2020.

A Review on Evolution of Architectures, Services, and Applications in Computing Towards Edge Computing



Pranay D. Saraf, Mahip M. Bartere, and Prasad P. Lokulwar

Abstract Over a decade, computing is the essential demand in World. Computing is the critical and integral component of modern generations of human life. User wants to manage, process, and communicate useful information in quick, efficient, and reliable way. The generations of computing are based on data storing and processing architectures. There are mainly four generations of computing. The first and third generations are based on centralized architectures, viz. as mainframe computing and cloud computing, whereas second and fourth are based on distributed architectures, viz. as client–server based computing and edge computing. The architectures of Edge computing give rise to many industrial applications including IoT, Security, Automobile, etc. It provided machine to machine communication, thus being analyzed by Gartner. The analysis gives statistics as around 75% of enterprise-generated data will get create and process outside traditional centralized data center or cloud, nowadays which is about 10%. This statistics gives rise to analyze the functionality, limitations and application areas of edge computing.

Keywords Computing · Edge · Cloud · Architecture · Services

1 Introduction

Computing alludes to any activity that is getting manage, process, and communicate information with the uses of computers. Computing specifically deals with processing of Data/Information in a quick and efficient way. In modern industrial technology,

P. D. Saraf (✉) · M. M. Bartere

Department of Computer Science and Engineering, G H Raisoni University, Amravati, India
e-mail: pranay.saraf@raisoni.net

M. M. Bartere

e-mail: mahip.bartere@raisoni.net

P. D. Saraf · P. P. Lokulwar

Department of Computer Science and Engineering, G H Raisoni College of Engineering, Nagpur, India
e-mail: prasad.lokulwar@raisoni.net

Fig. 1. 4 Generations in computing [21]



nowadays it is a critical, integral component. It incorporates using development of both hardware and software. Over a decade, computing is the essential demand in World. The era of computing starts with centralized computing in 1960s, as Mainframe computing, where powerful computers used for large information processing such as census, financial transactions, etc. Ruling over 2 decades, in 1980s, the evolution of distributed computing in the mode of client–server based models came into the picture. This evolution of the computing helps giant IT companies like Google, Yahoo, Alibaba, and many more (Fig. 1).

It was clear that the Distributed architectures are better than centralized in order to compute and process the operations with high speed. Thus, with collaborative efforts and considering features of various computing during the end of twentieth century and start of twenty-first century, a new model of Distributed Cloud Computing is evolved as Edge Computing. It includes computing ideas which were started from 1990 to 2016 including content delivery network (CDN), Pervasive Computing, peer-to-peer, Cloud Computing, Cloudlets, and Fog Computing. The trend of Edge computing is again going as ‘Back-to-Edge’ [10] as distributed architecture. These advanced/individual datacenters in new contexts: a datacenter in the sky (drones) perhaps even a datacenter on wheels (self-driving cars) [1, 8]. Thus, Edge computing provides a Backbone to Internet of Things (IoT).

2 Generations of Computing Architectures

2.1 *Mainframe Computing*

The Mainframe computing was the primary era of computing which is based on server. Its name mainframe is based on its construct as all units (handling, communication, etc.) were placed into one frame. It incorporates the capability to handle isolated processes of serialization, catalogs, program management, task management, job management, inter address space, and communication [4]. Some early and famous mainframes includes (Table 1):

Table 1 Top mainframe computers

Computer name	Year	Manufactured/Used by
Electronic Numeric Integrator and Calculator (ENIAC)	1942	Army Ordnance to compute World War II ballistic firing tables
MarkI	1944	US Army
Binary Automatic Computer (BINAC)	1949	Northrop Aircraft Company signs a contract with Eckert and Mauchly's company Electronic Control Company
Whirlwind	1960	air defense of the northern hemisphere During cold war between the USA and its allies and the former Soviet Union
Universal Automatic Computer (UNIVAC)	1952	Eckert-Mauchly Computer Company, was purchased by Sperry-Rand
IBM 701	1953	IBM, first production-line electronic digital computer from IBM with 1 Kb of RAM
IBM 360	1963	IBM provided with broad-based computing with standards
H-200 (Honeywell-115)	1969	US Air Force
Amdahl 470 V/6	1975	NASA
IBM 3090	1985	IBM
IBM S/390	1999	IBM introduced with Cryptographic Coprocessors

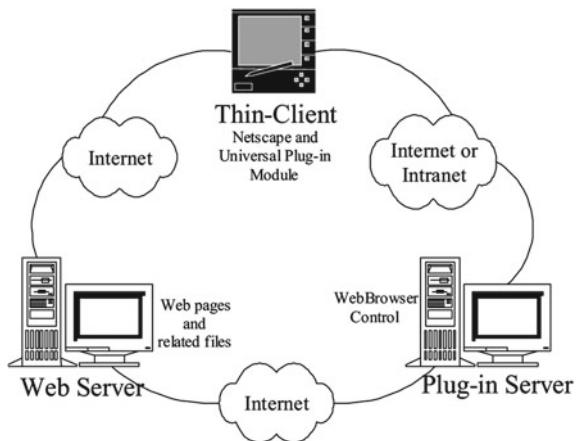
2.2 Client–Server Computing Model

The model of Client–Server computing is based on distributed computing model in where communication between client and server computers is taken care by a network. In client/server computing, client computers gives requests to server and on accepting requests, server shares its resources, applications and/or information with one or more client computers on the network. Xerox PARC, in 1970s, was the first client/server computing model introduced till today's exceedingly progressed client server computing systems [12].

The servers may include various types such as file servers, web servers, and mail servers. The client devices such as desktop computers, laptops, tablets, and smartphones will be using these services. Those single server can give resources to numerous clients at one time, thus may have a one-to-many relationship with clients [13]. When a client demands a connection request to a server, depending on the overload, the server can either accept or reject it (Fig. 2).

This architectural model is also called as two-tier architecture. In any case, if we revealed a few impediments of this design, failure of the server may lead to inoperability of the entire framework; it requires the high level of specialized staff; high cost of hardware [17]. We made the conclusion that multi-tiers architectures

Fig. 2 Client–server architecture [6]



can solve the issue of synchronization of version of applications and overcome most of the considered impediments [14].

2.3 Cloud Computing

The Cloud computing provides facility to storing and/or accessing data/information over the internet instead of locally on secondary storage of computer. The three terms you may hear associated with Cloud are:

(SaaS) Software as a Service—This model is based on a software licensing and delivery in which software is authorized on a membership premise and is centrally hosted.

(PaaS) Platform as a Service—This model provides as a category to cloud computing services. PaaS gives a platform permitting clients to create, run, and manage applications without the complexity of building and maintaining the infrastructure regularly related with creating and propelling an application [15].

(IaaS) Infrastructure as a Service—This model is also known as on-the-line computing. With Internet-based computing, IaaS provides shared processing resources and information to computers and other devices on demand of the client.

Cloud computing act like resource for data sharing and can be utilized incidentally too and it is very cost-effective, since the payment is based on client's utilization [19]. With prerequisites of standard web browsers and fast internet connections, they offer adaptability to get right away on demand to the client by sharing its pool assets to client web pages or IP [16].

A few of the breakthroughs within the history of cloud computing incorporate the dispatch of various services chronologically as Salesforce (1999), Amazon Web Services (2002), Linked-In (2003), Facebook (2004), Twitter (2006), Drop Box (2008), Google begun advertising browser based applications (2009), and i-Cloud

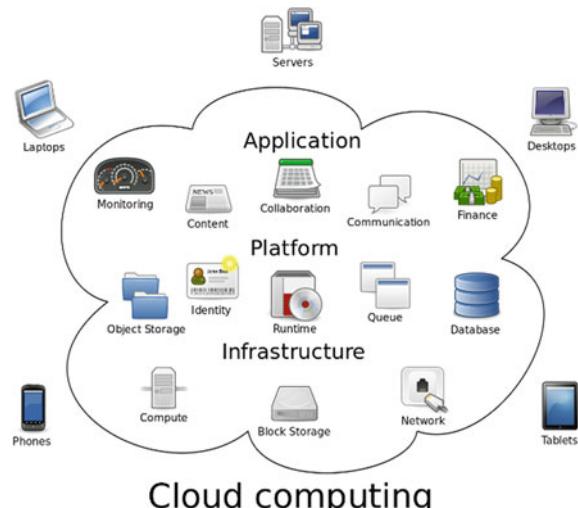
Table 2 Top cloud platform of 2019 by garter [5]

Cloud platform	Owned by
Amazon Web Services	Amazon Web Services
Microsoft Azure	Microsoft
Google Cloud Platform	Google
IBM Cloud	IBM
VMware Cloud on AWS	Vmware
Alibaba Cloud—International	Alibaba Cloud
Sales Cloud	Salesforce in Sales Force Automation
Oracle Cloud Infrastructure (Gen 2)	Oracle
Tencent Cloud—International	Tencent Cloud
Oracle Private Cloud	Oracle
vCloud Air powered by OVH	OVH

(2011) was presented. The blast of cloud based applications and services proceed to create at a disturbing rate. Following are top Cloud Platforms of 2019 ranked by Gartner (Table 2).

Gartner Survey about enterprise-generated data says, currently around 10% of this data is created and processed outside cloud or a traditional centralized data center. It also predicts that by 2022, this figure will reach 75% (Fig. 3).

Eventually, the cloud is utilized in any service that permits you to see data whether or not it has been spared locally. Without the utilization of cloud capacity and cloud innovation; all of our gadgets would run out of space exceptionally rapidly.

Fig. 3 Cloud architecture [9]

2.4 Intermediate Computings Before Evolution to Edge Computing

In between these computing era is moving towards Edge Computing, many forms of intermediate computing with its years of invention including Content Delevery Network (CDN) (1990), Pervasice Computing (1997), Peer-to-Peer (2001), Cloud Computing (2006), Cloudlets (2009), Fog Computing (2012) [7] (Fig. 4).

A content delivery network (CDN) alludes to group different servers which works together to supply quick delivery of content available over the internet, but those servers are geographically disseminated. The notoriety that CDN serves the larger part of web traffic, services proceeds to develop, and counting activity from major service applications like Amazon, HotStar, Netflix, Instagram and Facebook.

Pervasive computing, also known as ubiquitous computing is providing with the finest Internet capabilities, it may include artificial intelligence (AI) and voice recognition [4]. Moreover, pervasive computing applications has progressively been examined completely in different information areas and it is considered as a stepping stone for Information and Communications Technology (ICT) [22].

The Peer-to-Peer (P to P) computing architecture contains nodes which is considered as equal participants in data sharing. On these nodes, all the tasks which are required to comply are divided equally.

A cloudlet is developed with the goal is to “bring the cloud closer”, cloudlet can be viewed as a “data center in a box”. Cloudlet architecture provides cognitive assistance, with the low latency and rich computing power, including augmented reality and virtual reality that coherently enhance a user’s ability to interact with the real world around them [7].

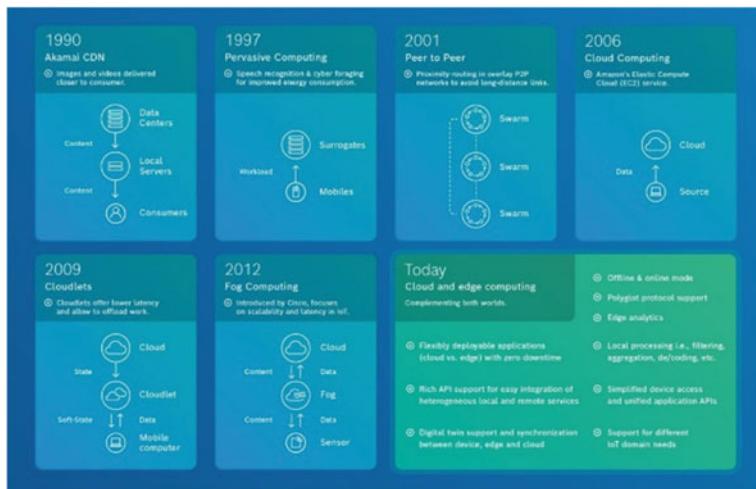


Fig. 4 Generation of computing before evolution of Edge Computing [1]

Fog computing is the concept of generation of network fabric that stretches from the outer edges. At those edges, data gets created and eventually get stored, either on cloud or customer's data center [20]. Fog has strong association with cloud computing and Internet of Things (IoT), which works as another layer of a distributed network environment. With Public Infrastructure as a Service (IaaS), cloud vendors can be thought of creation and storing of data from IoT devices as a high level, since the global endpoint for data may be considered as the edge of the network [7].

With such evolutionary concepts, the increase of data size goes from files to Big Data. Since, users want information on fingertips, the integrations of these computing proposed an architecture based on distributed computing as Edge Computing.

3 Edge Computing

Edge Computing (EC) is based on paradigm of distributed, open IT architecture in computing. With great capacity and features like decentralized processing power, enabling mobile computing and Internet of Things (IoT) technologies, it is considered as future of computing.

3.1 Architecture

In 2017, Edge Computing Consortium proposed an architecture of Edge Computing. The principles behind designing of the reference architecture is Model-Driven Engineering (MDE). The Edge Computing Consortium (ECC) proposes following Edge Computing Reference Architecture 2.0 (Fig. 5).

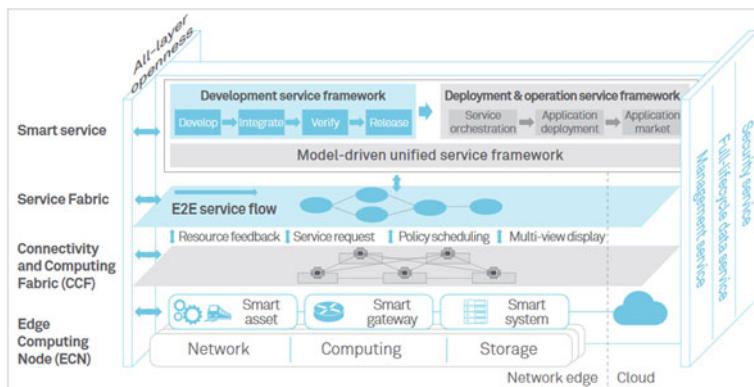


Fig. 5 Reference architecture for Edge Computing 2.0 proposed by Edge Computing Consortium in 2017 [3]

Service agility is provided with smart service coordination characteristics of E2E service stream through service fabric (SF) [10, 11] (Table 3).

Table 3 Summary of features provided by edge computing [2]

Computing characteristics	Status of edge
Latency	Low
Bandwidth utilization	Very low
Response time	Low
Storage	Low
Server overhead	Very low
Energy consumption	Low
Network congestion	Low
Scalability	High
Quality of service and quality of experience	High
Computing services	Status of edge
Computing service	Response time in milliseconds
Storage service	Temporary storage,
Doesn't Support Huge Data Collection	
Application	Status of edge
Road Safety	Available
Parking	Available
Traffic Signals	Available
Services	Status of edge
Network as a Service (NaaS)	Possible
Storage as a Service (STaaS)	Possible
Cooperation as a Service (CaaS)	Possible
Computing as a Service (COaaS)	Possible
Security challenges	Status of edge
Authentication	High Challenge
Vehicular Communication	Less Challenge
Localization	Less Challenge
Standards of computing provided by edge computing	
Dedicated Application Host at Edge Server	
Embedded OS at Edge Server	
Device management at the Edge of the Network	

3.2 Challenges

Edge computing is very different from traditional data center environments. Thus Edge Represents a Unique Computing Challenges:

1. **Security:** Security could be a foundational thought. This incorporates secure communication from the datacenter to the Edge, guaranteeing the security of information both at rest and in movement—anonymizing sensitive client information stored at the Edge.
2. **Availability and reliability:** Edge computing arrangements should be able to always monitor the health of the gateway runtime, distinguish deficiencies, and execute actions like reboot, restart, or initiate a factory reset.
3. **Remote management and update:** It should be possible to remotely manage edge computing devices to start, stop, configure the device, and remotely install new software and hotfixes.
4. **Analytics and machine learning:** Real-time data processing and its analytics is still a challenge in Edge computing. An edge computing arrangement should make it conceivable to apply distinctive machine learning algorithms at the edge.
5. **Open API for future applications:** Edge computing can communicate with the Internet of Things (IoT) due to its faster speed. The role of Edge computing as a crucial component for making sure that 5G can work seamlessly.

3.3 Application

By conveying edge intelligence services, EC meets the key necessities of industry digitalization for agile connectivity, security and privacy protection, application intelligence, real-time administrations, data optimization.

3.3.1 Use Cases on Edge Computing by Bosch [1]

The wise range of industries are putting use cases on Edge computing. The following use cases give some additional understanding to how edge computing can solve some of the challenges of IoT deployments:

Industrial IoT:

There are numerous use cases available for Industrial IoT which require quick response times enabled by edge computing.

- For security-based systems, immediate reaction time required to maintain a strategic distance from physical injury and machine damage in case human entered in restricted area. In this case, use of edge computing is critical as to the capacity to

INDUSTRIAL IoT DATA PROCESSING LAYER STACK

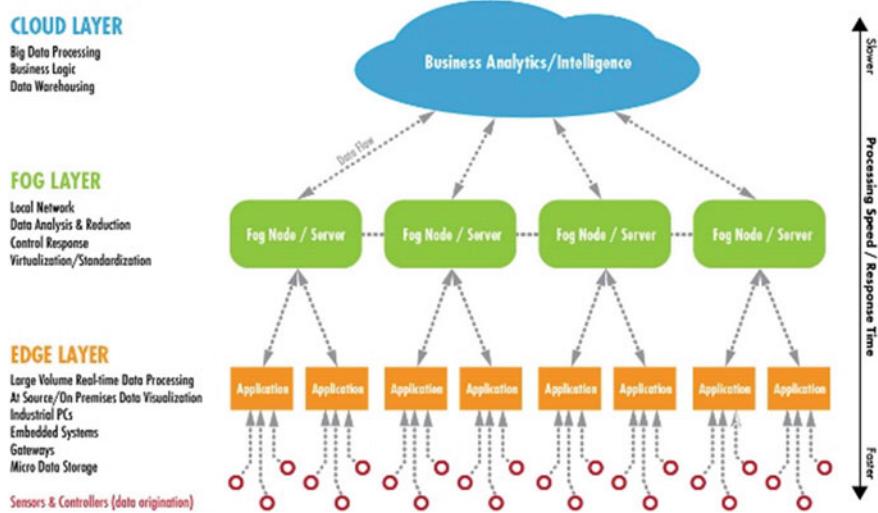


Fig. 6 Data Processing Layer stack between Cloud, Fog, and Edge Layer [18]

make quick, deterministic decisions for human security and machine conservation in industrial IoT applications (Fig. 6).

Retail

Supermarkets and retail stores can advantage from IoT applications to improve the productivity of their buildings and client experience. A few examples how edge computing makes a difference retailers:

- An edge computing can provide solution for reverse vending machine to send employees a push notification once the machine becomes 80% full. This would allow the employee plenty of time to empty the machine before a potential negative customer feedback of being delayed by a full machine [21, 23].

3.4 Future Trends and Opportunities in Edge Computing

- **5G:** With the tremendous growth of digitalization over all the industries—5G could be one of the key technology. (Mobile) Edge computing may be a significant portion of the 5G platform and gives the foremost advantage of communication benefit to suppliers in modern business opportunities to grab the opportunities [23].

- **Gaming:** Internet gaming is as of now a favorite interest for people. The gaming communities across the globe is focusing on this as a business opportunity. It is predicted by Cisco that, the internet gaming traffic will increase nine times of 2017 by 2022. This signifies that internet gaming will account for 4% of IP activity by 2022 as of 2017.
- **Artificial Intelligence and Machine Learning:** High performance computing (HPC) with its demanding applications including AI/ML workloads edge data centers are the only culminate vehicle. It has high levels of computing power with smaller physical impressions [7].
- **Streaming Wars:** Various Content providers are much dependent on content distribution networks (CDNs) to cache information sensibly near to end-users. Globally, the demand of video traffic will account for 82% of all trade and consumer IP activity by 2022, predicted by Cisco. At the same time, worldwide use of VR/AR traffic will develop/increase 12 times between 2017 and 2022.

4 Conclusion

The era is fast, computing started as the user wants information on fingertips. Thus, the architectures which provides such information need to upgrade as per the requirements. For the last a few years, cloud computing is the focused area for enterprises due to its developing methodologies to “move to the cloud” or at least “expand into the cloud.” It’s been considered as the only solution. To move ahead for low latency and real-time processing with location and distributed processing, where we ought to grow our thinking beyond centralization and cloud. The Edge computing gives the paradigm shift from centralize to distributed again. The International Data Corporation (IDC) predicted that by 2020, spend on edge infrastructure in IT will reach up to 18% of the total spend on IoT infrastructure. This is driven by deployments of converged Information Technology (IT)/Operation Technology (OT) systems that reduce the time to value of data.

References

1. Edge computing for IoT-A guide on how edge computing complements the cloud in IoT, in *Bosch Software Innovations GmbH* (December 2018)
2. H. El-Sayed et al., Edge of things: the big picture on the integration of edge, IoT and the cloud in a distributed computing environment. *IEEE Access* **6**, 1706–1717 (2018)
3. Edge Computing Consortium (ECC) and Alliance of Industrial Internet (AII). *Edge Computing Reference Architecture 2.0* (November, 2017)
4. J. Sen, *Ubiquitous Computing: Applications, Challenges and Future Trends Chapter 1*. <https://doi.org/10.1201/b12298-2> (2012)
5. Gartner survey on Cloud Infrastructure as a Service, Worldwide (2019). <https://www.gartner.com/reviews/customers-choice/public-cloud-iaas>

6. C.-C. Kuo, P. Ting, W.-G. Teng, P. Chen, M.-S. Chen, J.-C. Chen, Multimedia over IP for thin clients: building a collaborative resource-sharing prototype. *Concurr. Eng. R&A* **12**, 175–183 (2004)
7. K. Dolui, S.K. Datta, Comparison of edge computing implementations: fog computing, cloudlet and mobile edge computing, in *2017 Global Internet of Things Summit (GIoTS)* (Geneva, 2017), pp. 1–6
8. H. Bangui, S. Rakrak, S. Raghay, B. Buhnova, Moving to the edge-cloud-of-things: recent advances and future research directions. *Electronics* **7**, 309 (2018). <https://doi.org/10.3390/electronics7110309>
9. Cloud Computing. https://en.wikipedia.org/wiki/Cloud_computing
10. S. Sarkar, S. Chatterjee, S. Misra, Assessment of the suitability of fog computing in the context of internet of things, in *IEEE Transactions on Cloud Computing*, vol. 6, no. 1 (Jan.–March 2018), pp. 46–59
11. S. Medhat, Client/server computing—an engine for change and growth, in *International Seminar on Client/Server Computing. Key Note Addresses, La Hulpe, Belgium*, , vol. 2 (1995), pp. 1/1–120
12. S.P. Savino, S.M. Queroli, Impact of client/server computing on the telecommunications industry, in *IEMC 96 Proceedings. International Conference on Engineering and Technology Management. Managing Virtual Enterprises: A Convergence of Communications, Computing, and Energy Technologies* (Vancouver, BC, Canada, 1996), pp. 605–610
13. R.L.R. Maata, R. Cordova, B. Sudramurthy, A. Halibas, Design and Implementation of client-server based application using socket programming in a distributed computing environment, in *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIIC)* (Coimbatore, 2017), pp. 1–4
14. Moving the Cloud to the Edge. <https://www.pubnub.com/blog/moving-the-cloud-to-the-edge-computing/>
15. Y. Amanatullah, C. Lim, H.P. Ipung, A. Juliandri, Toward cloud computing reference architecture: Cloud service management perspective, in *International Conference on ICT for Smart Society* (Jakarta, 2013), pp. 1–4
16. M. Bahrami, Cloud computing for emerging mobile cloud apps, in *2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering* (San Francisco, CA, 2015), pp. 4–5
17. M. Aazam, E.N. Huh, Inter-cloud media storage and media cloud architecture for inter-cloud communication, in *2014 IEEE 7th International Conference on Cloud Computing* (Anchorage, AK, 2014), pp. 982–985
18. J. Yang, L. Zhang, X.A. Wang, On cloud computing middleware architecture, in *2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)* (Krakow, 2015), pp. 832–835
19. S.S. Adhatarao, M. Arumaithurai, X. Fu, FOGG: a fog computing based gateway to integrate sensor networks to internet, in *2017 29th International Teletraffic Congress (ITC 29)* (Genoa, 2017), pp. 42–47
20. N. Hassan, S. Gillani, E. Ahmed, I. Yaqoob, M. Imran *The Role of Edge Computing in Internet of Things*. *IEEE Communications Magazine*, vol. 56, no. 11 (2018), pp. 110–115
21. Back to Edge, blog on Nautix Next, available on <https://next.nutanix.com/blog-40/back-to-the-edge-27879>
22. Y. Cao, S. Chen, P. Hou, D. Brown, Fast: A Fog Computing assisted distributed analytics system to monitor fall for stroke mitigation, in *2015 IEEE International Conference on Networking, Architecture and Storage (NAS)*, (IEEE, 2015), pp. 2–11
23. *Cloud, Fog and Edge Computing—What's the Difference?* December 4, 2017. <https://www.winsystems.com/cloud-fog-and-edge-computing-whats-the-difference/>

Image Retrieval Using Multilayer Bi-LSTM



Shaily Malik, Poonam Bansal, Pratham Sharma, Rocky Jain,
and Ankit Vashisht

Abstract The growth of the internet and the social media has led to rapid growth of the digital images available online as digital databases. The problem of searching such databases for items that are similar to a query image or a query descriptor is termed as image retrieval, and such systems are called image retrieval systems. An image retrieval system is a computer system that helps in looking up a large database of digital images and retrieving the images, which matches a user's request. These systems are used for retrieving images related to the user request from the database. The following research will state the role of generating descriptions for images in these systems and proposing the use a multilayer biLSTM (bidirectional LSTM) for the purpose of generating descriptions. The proposed model gave desirable results, which were also better results than its counterparts, i.e., LSTM and biLSTM.

Keywords Image retrieval · Deep features · Multilayer biLSTM · Description generation

1 Introduction

An image retrieval system is a computer system that helps in looking up a large database of digital images and retrieving the images, which matches a user request [1]. An ideal image retrieval system is one that returns images that are more similar to the query. This is a challenging task as the system must be trained on recognizing the exact context of the images. Image retrieval using a text query is used in the majority of image search systems, which depends on the metadata of the images and relies on human input and is, thus, prone to errors.

The process of image retrieval is threefold. The process begins with the user's attempt to query the database. The query may be a complete picture, a description, a part of a picture, or may even be a complex query wherein the user selects different sections from different pictures, which are later combined to form a single query.

S. Malik (✉) · P. Bansal · P. Sharma · R. Jain · A. Vashisht
Computer Science and Engineering Department, Maharaja Surajmal Institute of Technology,
GGSIPU, New Delhi, India

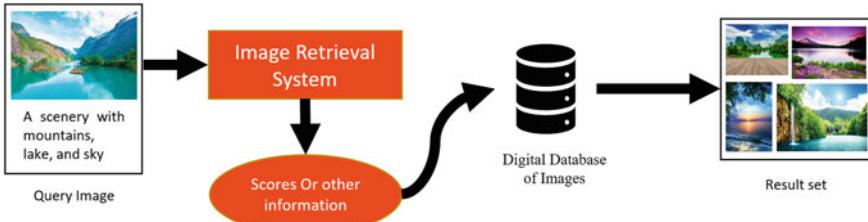


Fig. 1 A typical image retrieval system

This query is then processed by the image retrieval system, the system evaluates the query based on specified attributes (color, shape, etc.) using various techniques. The results are then aggregated to find images from the database, which are similar to the query images. Finally, these images are returned as the result to the query (Fig. 1).

Many image retrieval systems use one or multiple images as query to get similar images. The current image retrieval systems in such cases use features such as color and shape for fetching the results, but the results are not as close to the query as desired as it is hard to get the context of an image primarily on the basis of its shape, size, and color. Hence research to address this problem is necessary.

Several recent research in the field of deep learning have made great strides and have helped machine learning to surpass human performance. The deep architecture incorporates low-level features into high-level non-visual features with non-linear transformation, allowing it to be able to learn semantic representation from images [2]. As a result of such advancements in deep learning and processing power, there have been a growing number of research in image retrieval by generating description.

Descriptions help us to address this problem to get the context of the input query and retrieve similar images, which otherwise was a challenge while using shape and color knowledge to retrieve images.

The rest of the paper is organized as follows: Section 2 discusses several image retrieval methodologies that were developed as a result of previous researches, Section 3 discusses in detail the proposed methodology and how it is implemented. Section 4 discusses the experimental results and Section 5 concludes research work.

2 Related Work

Image retrieval systems have been around for a long time and a number of researches have been done in this field, and most involve the concept of feature extraction. For this, several techniques are employed such as content-based indexing [3], composition of regions using similarities [4, 5], aggregating features and descriptors from images [6, 7], etc. [8] show the importance of feature selection and extraction and how this can help you achieve great results. These techniques, many times, also use deep learning for improved results. In the present scenario, as a result of the progress made

by convolutional neural networks (CNNs), they have become the primary choice for feature extraction in recognition tasks [2, 3, 6, 7, 9].

Several methodologies have been adapted in the past for image retrieval, hashing methodologies have been adapted by Lu and Huang et al. [10, 11]. Lu et al. [10] propose hierarchical recurrent neural hashing (HRNH), a hashing mechanism that makes use of hierarchical recurrent neural network to generate similar hash codes for similar images, which will facilitate the task of retrieval. Similarly, Ref. [11] proposes a deep architecture to generate a binary mask that can approximately identify locations of foreground objects in an image, further processing of these masks helps in similarity calculation. Wang et al. [12] use a tag-based image retrieval approach by extracting global and local features. Song et al. [13] propose a method to improvise over the results of CNN for the purpose of semantic-based image retrieval. Yang et al. [14] use calculation of pairwise distances between the deep features for the query and candidate image.

Another class of image retrieval system is content-based system (CBIRs) that uses features of image objects such as color, shape, size, gradients, edges, etc. to search for similarities between images rather than annotation or tagging techniques. Omhover et al. [4] propose an image retrieval system, which uses fuzzy similarity measures derived from psychological considerations to evaluate regions of a segmented image to region visual similarity scores based on the color and shape of the region. Gordo et al. [7] propose an effective and scalable approach on instance-level image retrieval by producing a fixed-length representation for each image by aggregating region-wise descriptors. Babenko and Lempitsky [6] proposed sum-pooled convolutional features (SPoC descriptors), which is based on the aggregation of raw deep convolutional features extracted using the very deep CNN [15].

3 Methodology

The process of generating descriptions in most cases is performed using deep learning models and can be called a two-step process consisting of 1. Feature extraction and 2. Description generation. These steps here are achieved using convolution neural networks (CNN) and long short-term memory (LSTM) as shown in Fig. 2.

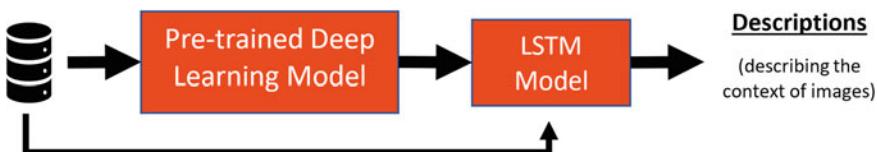


Fig. 2 Proposed network for generating descriptions

3.1 Dataset

The dataset used should be such that it is able to train the proposed model for generating keywords for a query image, which could be used to retrieve images. Thus, for this research, we would use the Flickr8k dataset [16], which comprise of 8000 different images taken from the Flickr website, and each image is mapped to five different sentences describing the image. The dataset represents different events and scenarios and does not comprise images having well-known places and people, which makes the dataset more non-specific. The dataset is split into 6K training images, 1K images in development dataset, and 1K test images. Once trained on the Flickr8k dataset, this model can be extended to any dataset for the purpose of retrieval. Here we would extend this to Wang's image dataset [17, 18] and compare our results with some previous works. The Wang's image dataset has 1K images with 10 classes having 100 images each, these would be split into 90% train images, which would be used as our database of images, and 10% test images used to calculate the accuracy and scores.

3.2 Data Pre-Processing

Before we can use our dataset for training purposes, a series of steps shall be carried out to transform the raw data into meaningful and useful format, these include cleaning, reducing, and transformation of the data. The dataset used has five descriptions associated with each image, which must be cleaned before these could be fed into the network. The cleaning process involves removal of all punctuations, case conversion (lowercase here) to ensure uniformity, and adding flags to show the starting and ending of a sequence to LSTM (for this we will append words 'startseq' and 'endseq' at starting and ending of the sequence respectively). Additionally, to create vocabulary of the dataset, the clean descriptions of images are used to form a set of unique words.

3.3 Feature Extraction

Feature extraction is a useful method in image processing that is used to reduce the resources needed to process the given data without losing important or relevant information. It is related to dimensionality reduction, which aims at creating new features from existing ones discarding redundant and useless information, and ultimately reducing the amount of data for analysis. Feature extraction is the combined name for all the methods that help effectively and accurately reduce the amount of data that needs to be processed. This helps in reducing the time required for processing, prevent overfitting, and improves accuracy [19].

For our research work, we will be using the deep convolution neural network (DCNN) VGG19 to extract the deep features for a given image, which would then be later used for generating description. The deep feature extraction is a method used in machine learning for solving different problems by using the knowledge gained from the solution of a previous problem [20]. The architecture created as a solution to a problem can be used to assist the new architecture that is intended to solve a new problem.

The VGG19 (Visual Geometry Group) [15] is a 19-layer deep convolutional neural network. It consists of 16 convolution layers, 3 fully connected layers, 5 MaxPool layers, and 1 SoftMax layer, of which only the convolution and maxpool layers have trainable weights. VGG19 is pretrained on more than a million images from the ImageNet [21] database and is capable of classifying images into 1000 object categories. The network has learnt rich feature representations for a wide range of images. VGG is the successor of the AlexNet and has 19.6 billion FLOPs (floating-point operations) to achieve more computational efficiency. VGG uses kernels of (3×3) size with a stride size of 1 pixel and spatial padding to preserve the spatial resolution of the image. Thus, for this research, we will be using this to extract the deep features for the images. The architecture of the VGG19 consists of two parts, the first part is responsible for feature extraction using a CNN and the second part consisting of fully connected and softmax layer does the classification based on these extracted features.

For acquiring deep features, the CNN part of VGG19 with pretrained image net weights is used. The model takes an image input of size 224×224 and gives deep features of dimension 1×4096 , which are simply reduced representation of the images. These features are used to evaluate performance using the multilayer biLSTM model.

3.4 Generating Descriptions

Once the deep features for the image are extracted, we can now proceed to work on generating descriptions using these extracted features. This is one of the most crucial parts of the entire process. The descriptions represent the context of the image as perceived by the system. The process that we use to predict these descriptors should be able to predict the most likely meaning of a given image. Irrelevant words would result in less similarity between the query and result images. Thus, for this purpose, we will use a LSTM model, which would be trained to generate the most appropriate descriptions for a given image. LSTMs can recall the previous inputs for long time as they are saved into the memory, and thus can be employed in problems with sequences having long gaps.

The conventional LSTM, however, comes with several shortcomings because of which there exist two variations of LSTM, namely, multilayer LSTM and biLSTM. In case of multilayer LSTM (or sometimes stacked LSTM), there are multiple LSTM layers with recurrent relation in between the units of the same layer, and feedforward

relation in between the units of an LSTM layer and the layer over it. The main reason for stacking LSTM is to allow for greater model complexity. Like in a simple feedforward neural net, we stack layers to make a hierarchical feature depiction of the input data. The same applies to stacked LSTMs. This hierarchy of hidden layers allows the depiction of more complex data and enables capturing information at different scales. Bidirectional LSTM or simply BiLSTM has a basic idea to present each training sequence forwards and backwards to two LSTM once from beginning to the end and once from end to beginning. By processing the inputs from both directions, it utilizes both the previous and future context. Using these, we can utilize long-range context in both the input directions.

Thus, for this research, we will combine the power of these two variations for the purpose of generating descriptions for image retrieval.

For generating descriptions that describe the context of a query, each description is split into words. The model will be provided a word and the image and will generate the next word. Then the first two words of the description will be provided to the model as input with the image to generate the next word. This is how the model will be trained.

Later, when we generate the descriptions using the model, the generated words will be concatenated and recursively provided as input to generate a description for an image. Since this would be a batch processing, we need to make sure that each sequence is of equal length. Hence, zeros are appended at the end of each sequence to make them the same length.

3.5 Training the Model

With all the data preprocessing completed and deep features extracted, we will now use both to train our multilayer biLSTM to generate the most optimal descriptions for an image.

The architecture of the model as shown in Fig. 3 consists of two networks running parallel to each other, one works with the textual part and the other with the images. As described before the dataset has five descriptive sentences associated with each

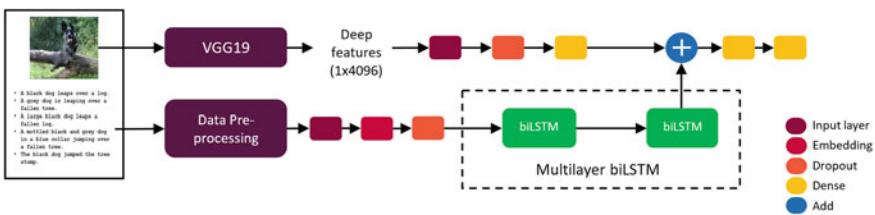


Fig. 3 Architecture for the proposed Multilayer biLSTM model

image, this set is broken down into five sets each having the image and one description. These sets are then fed into the network. The image in each set is passed through the pretrained VGG19 model, which gives us the deep features for the image. These are then passed through further layers. Alongside, the textual part of the set is cleaned, reduced, transformed, and encoded, before passing on to further layers. The network responsible to train on the textual data also has an embedding layer, which is initializing by random weights for all words and later learns the embeddings for all the words in the dataset. The results are then passed on to the multilayer biLSTM model whose output is then combined with the output from the first network to get the actual desired results from the dense layers.

The training of the network was achieved with Adam learning method and categorical cross-entropy as the loss function and was carried out on the free version of Google Colab, a product from Google research that allows free access to computing resources including GPUs. The GPUs available in Colab often include Nvidia K80s, T4s, P4s and P100s that come with about 12 GB RAM and 68 GB disk space.

3.6 Retrieving Images

With a database of images having descriptions available for all, we can now fetch images from it corresponding to a query image. The pretrained VGG19 model is first used to extract the deep features for the image the same way it was done during the training process. These deep features are then used to generate a description for the query using the trained multilayer biLSTM model. Alongside, a Bag of Words (BoW) consisting of vectors of dimension $1 \times$ vocabulary size corresponding to each image from the data is formed using the descriptions available as a part of the database. The descriptions that were obtained from the multilayer biLSTM model have also converted to a vector of similar dimension. A dot product of this vector is then performed with each vector from the BoW and the images corresponding to the top K results are returned as the result set for the query.

3.7 Evaluation and Scoring

We will be using precision, recall, and F1 score to evaluate and check the performance of the retrieval system. Precision recall and F1 help you evaluate the false-positive rates, false-negative rates, and sensitivity of your model. We will try to achieve a desirable precision for individual classes as well as a desirable average precision. For this, we would need the information about true positives, true negatives, false positives, and false negatives for the query. Precision gives us a measure of the relevant data points. In our case, precision can be calculated as the ratio of relevant images retrieved to the total images retrieved.

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \quad (1)$$

The recall, on the other hand, is the measure of our model correctly identifying true positives. It gives a measure of how accurately our model can identify the relevant data. We refer to it as sensitivity or true positive rate. For our case, recall would be the ratio of number of relevant images retrieved to the total relevant images that were present.

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \quad (2)$$

Thus, all the scores could be calculated individually for all the images in the result set and these individual results could later be combined to calculate the overall result. We will then compare the best results with the results from a LSTM and a biLSTM.

4 Result and Discussion

Precision and recall are statistical measures of the relevance of a retrieved image with respect to the query. For our research, these are calculated as:

$$\text{Precision} = \frac{\text{no. of relevant images retrieved}}{\text{total no. of images retrieved}} \quad (3)$$

$$\text{Recall} = \frac{\text{no. of relevant images retrieved}}{\text{total no. of images retrieved}} \quad (4)$$

Figure 4 shows the class-wise precision and recall values for $k = 10$ images retrieved per query.

Figure 5 shows the precision–recall graph for $k = 10$ images retrieved per query.

Fig. 4 Class-wise precision and recall values for $k = 10$

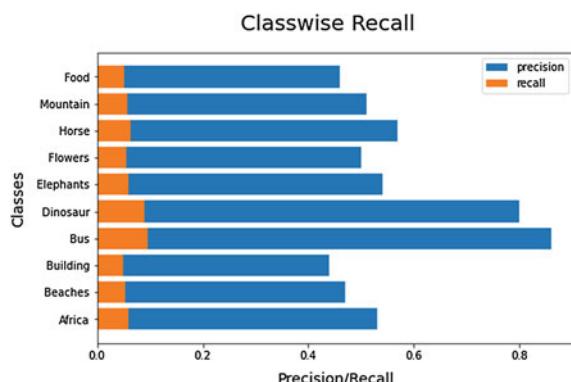


Fig. 5 Precision–Recall graph for $k = 10$

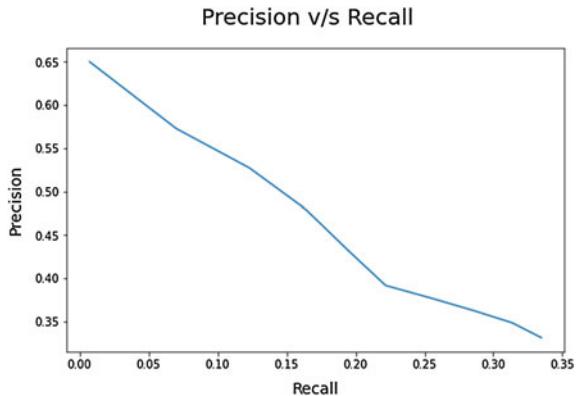


Fig. 6 Average precision for different number of images retrieved per query

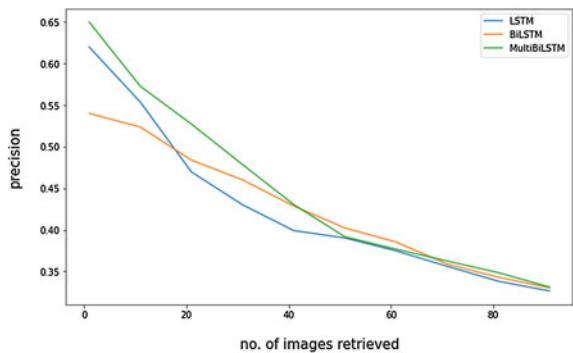


Figure 6 shows the overall precision of a different number of images retrieved per query. The value varies from $k = 10$ images to 100 images per query.

This would help us chose the optimal number of images to be retrieved from the database for a given query. The selection of k shall take two things into consideration, first that the value of k should be as large as possible, and second that the precision should not go below a desired value. From the figure, we can clearly see that the precision decreases as the number of images retrieved increases in number indicating that the relevance of images retrieved decreases as k increases for a query. From the plot, we can see that a multilayer biLSTM gives better results than a LSTM and a biLSTM till $k = 40$ after which all values become almost same; we can also make out that 10–20 images retrieved per query would an optimal number for our system.

Table 1 shows a comparative analysis of class-wise precision of the proposed model with the precision obtained using a LSTM and a biLSTM.

Table 1 Comparison between precision obtained using the proposed model, LSTM and biLSTM

Class Name	LSTM	biLSTM	Multilayer biLSTM
Africa	58	55	53
Beaches	61	54	47
Building	64	48	44
Bus	82	63	86
Dinosaur	87	90	80
Elephants	20	43	54
Flowers	44	41	50
Horse	46	52	57
Mountain	55	52	51
Food	47	30	48
Average	56.4	52.8	57

5 Conclusion and Future Scope

Content-based image retrieval systems have gained a lot of research attention considering their application benefits. This paper introduces a mechanism for image retrieval with the major consideration of using a multilayer biLSTM model to generate descriptions of images, which could then be used for the purpose of retrieval. The model was successfully trained to serve the purpose of image retrieval from a digital database of images using image encodings generated with the help of VGG19 and their corresponding descriptions available as a part of the dataset. The model was tested on the Wang's image dataset, and a decent class wise precision was obtained. The proposed model better results for object-based images than scenery-based images with the highest precision of 86% for bus class. These results were also better than that from the LSTM (which gave better results for scenery-based images) and biLSTM.

The multilayer biLSTM model is a quite complex network and could give much better results for datasets larger than the Flickr8k dataset. The proposed model for image retrieval could also be extended for a digital database of videos.

References

1. R.A. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval* (Addison-Wesley Longman Publishing Co. Inc., USA, 1999)
2. H. Wang, Y. Cai, Y. Zhang, H. Pan, W. Lv and H. Han, deep learning for image retrieval: what works and what doesn't, in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, Atlantic City, NJ, 2015, pp. 1576–1583. <https://doi.org/10.1109/ICDMW.2015.121>
3. I. Úbeda, J.M. Saavedra, S. Nicolas, C. Petitjean, L. Heutte, Pattern spotting in historical documents using convolutional models. [arXiv:1906.8580v1](https://arxiv.org/abs/1906.8580v1) [cs.CV] (2019)
4. J.F. Omhover, M. Detyniecki, Image retrieval by composition of regions. <https://doi.org/10.1016/B978-044452075-3/50036-4> (2006)

5. J. Fauqueur, N. Boujemaa, New image retrieval paradigm: logical composition of region categories, in *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*, Barcelona, Spain, 2003, pp. III-601. <https://doi.org/10.1109/ICIP.2003.1247316>
6. A. Babenko, V. Lempitsky, Aggregating deep convolutional features for image retrieval. [arXiv: 1510.07493v1 \[cs.CV\]](https://arxiv.org/abs/1510.07493v1) (2015)
7. A. Gordo, J. Almazán, J. Revaud, D. Larlus, Deep image retrieval: learning global representations for image search, in *Computer Vision—ECCV 2016. ECCV 2016*, Lecture Notes in Computer Science, vol 9910, eds by B. Leibe, J. Matas, N. Sebe, M. Welling (Springer, Cham, 2016). https://doi.org/10.1007/978-3-319-46466-4_15
8. R.J. Raj, S.J. Shobana, I.V. Pustokhina, D.A. Pustokhin, D. Gupta, K. Shankar, Optimal feature selection based medical image classification using deep learning model in internet of medical things. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2020.2981337> (2020)
9. A.S. Razavian, H. Azizpour, J. Sullivan, et al., CNN features off-the-shelf: an astounding baseline for recognition, in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2014)* (IEEE Computer, 2014)
10. X. Lu, Y. Chen, X. Li, Hierarchical recurrent neural hashing for image retrieval with hierarchical convolutional features. *IEEE Trans. Image Process.* **27**(1), 106–120 (2018)
11. C. Huang, S. Yang, Y. Pan, H. Lai, Object-location-aware hashing for multilabel image retrieval via automatic mask learning. *IEEE Trans. Image Process.* **27**(9), 4490–4502 (2018)
12. Y. Wang, L. Zhu, X. Qian, J. Han, Joint hypergraph learning for tag-based image retrieval. *IEEE Trans. Image Process* **27**(9), 4437–4451 (2018)
13. K. Song, F. Li, F. Long, J. Wang, Q. Ling, Discriminative deep feature learning for semantic-based image retrieval. *IEEE Access* **6**, 44268–44280 (2018)
14. J. Yang, J. Liang, H. Shen, K. Wang, P.L. Rosin, M. Yang, Dynamic match kernel with deep convolutional features for image retrieval. *IEEE Trans. Image Process* **27**(11), 5288–5302 (2018)
15. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
16. M. Hodosh, P. Young, J. Hockenmaier, Framing image description as a ranking task: data, models, and evaluation metrics. *J. Artif. Int. Res.* **47**(1), 853–899 (May 2013)
17. J. Li, J.Z. Wang, Automatic linguistic indexing of pictures by a statistical model-ling approach. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(9), 1075–1088 (2003)
18. J.Z. Wang, J. Li, G. Wiederhold, Simplicity: semantics-sensitive integrated matching for picture libraries. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(9), 947–963 (2001)
19. “Feature Extraction,” deepai.org, para. 1. <https://deepai.org/machine-learning-glossary-and-terms/feature-extraction>. Accessed 12 Nov 2020
20. M. Turkoglu, D. Hanbay, A. Sengur, Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests. *J. Ambient. Intell. Hum. Comput.* (2019). <https://doi.org/10.1007/s12652-019-01591-w>
21. J. Deng, W. Dong, R. Socher, LJ Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in *2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL* (2009), pp. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>

Analysis of FSO-OFDM System Performance for Different Bit Rates and Link Ranges



Nabadh Bhan and Sanmukh Kaur

Abstract With advancements in technology, optical communication has become a necessity for the world. The need for advancements in this field has increased more than ever. Free space optics (FSO) technology allows communication to be achieved wirelessly in free space, i.e. air, space, vacuum, etc. It allows the use of high bandwidth, security and good power transmission. The main disadvantage is that it requires line of sight, so any obstruction or turns in between the channel can lead to loss of data. It has been used for both, terrestrial and satellite applications in the past. The model that has been implemented in this work uses Orthogonal Frequency Division Multiplexing (OFDM) modulation scheme in an FSO channel employing 4-QAM (Quadrature Amplitude Modulation). The performance of FSO-OFDM model has been analyzed for varying bit rates and link ranges on Optisystem software. The system supports a data rate of 15 Gbps up to a link range of 13 km for clear weather conditions.

Keywords Optical Communication · Free Space Optics (FSO) · Orthogonal Frequency Division Multiplexing (OFDM) · Quadrature Amplitude Modulation (QAM)

1 Introduction

The project was undertaken to conduct a research on the FSO system and implement its model using an advanced modulation technique called OFDM.

The acronym “OFDM” means orthogonal frequency division multiplexing and is a variation of frequency division multiplexing (FDM) technique. Orthogonal frequency division multiplexing is a technique that has been in use since 1966 and has been used in various technologies like cellular systems, underwater communications, wireless LANs, etc. This work takes into consideration its use as an optical light modulation technique [1].

N. Bhan (✉) · S. Kaur

Amity School of Engineering and Technology, Uttar Pradesh, Sector 125, Noida, India

FSO permits optical connectivity acting as a medium between receiver and transmitter. FSO link is used for outdoor wireless communication [2]. In this paper, we present OFDM-FSO link.

The goal of this work was to investigate the performance of this system for various link ranges and data rates. It is also observed that how the system behaves at changing parameters such as power, attenuation constant, etc. Bit error rate (BER) at different bit rates and link ranges are compared for different weather conditions.

The reason OFDM scheme was chosen for this work was that it strengthens long-distance communication by reducing interference and improving signal to noise ratio.

Section 2 of the paper explains FSO-OFDM in depth. Section 3 explains the working and simulation of the system. Section 4 discusses the results of the experiment and Section 5 concludes the work.

2 FSO-OFDM

FSO, which stands for free-space optics, is a modern technology that was brought to improve wireless optical communications in the world. As the name suggests, “free space” here means empty space which could be space, air, vacuum, etc. The terms “fiber less” and “optical wireless technology” are associated with it as it is a technology that does not require any wire for the transmission of data. It uses light source such as lasers and photodiode as detectors for communication [3]. Several parameters are taken into consideration when FSO technology is used. They are classified into two categories—internal and external parameters. Internal parameters include factors like wavelength, bandwidth, BER, etc. External parameters include factors like visibility, weather conditions, line of sight, etc. [4].

Since FSO is concerned with optical communication, therefore, the transmission of data is in light form.

2.1 Challenges Faced by FSO

Alas, no technology is perfect. Ever evolving technology has its problems, as does free-space optical communication. Following are some of the main challenges faced in the FSO channel

- Line of sight: Since the communication is done via lasers, the receiver has to be placed in the LOS of the source due to the obvious fact that laser light cannot bend. So, in order for the data to reach the receiver end, it has to be placed where laser can directly reach it without any obstructions in between.

- Weather conditions: Avoid weather conditions such fog, snow, pollution and even rain that can cause loss in data as it can affect the field of view due to an increase in attenuation.
- Atmospheric turbulence: Fluctuating atmospheric temperature can cause variations in the refractive index of atmosphere, which can lead to change in amplitude and phase in free space [5–7].

OFDM stands for orthogonal frequency division multiplexing. It is an upgraded version of FDM, which is frequency division multiplexing and holds some advantages over it. The subcarriers in OFDM are usually modulated using QAM or PSK techniques. This technology is used in DSL Internet access, broadcasting, wireless networks, 4G mobile communications [8]. This project uses 4-QAM OFDM technique.

2.2 *Difference Between OFDM and FDM*

In FDM modulation, the subcarriers are independent and separate from each other. There are also guard bands between each subcarrier to avoid interference as shown in Fig. 1.

However, in OFDM modulation, the subcarriers overlap each other as shown in Fig. 2. As it can be seen, each subcarrier's peak is achieved when other subcarriers are at null. This way they can be distinguished from each other [9, 10].

Fig. 1 Frequency Division Multiplexing

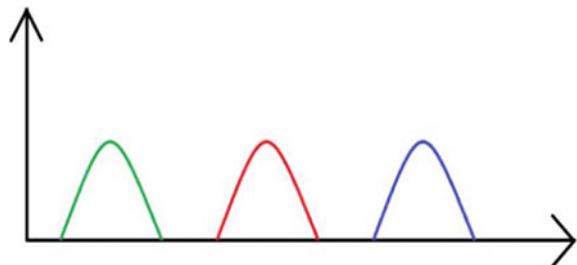
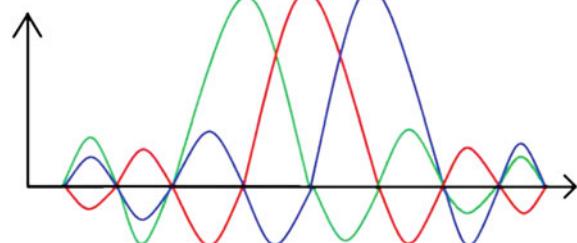


Fig. 2 Orthogonal Frequency Division Multiplexing



2.3 Advantages of OFDM

Since, the subcarriers overlap with each other and there is no need for guard band, OFDM is able to effectively utilize its bandwidth. Also, the signals can be distinguished from each other as peak of every signal occurs at null of the others, this way we can prevent interference of signals if we use this modulation technique. Due to such major advantages, this modulation technique is sometimes preferred over FDM [11, 12].

3 System Setup

Figure 3 shows the schematic diagram of the layout of the software.

This whole model is implemented on the software OPTISYSTEM 17.0. It is software used to implement and analyze modern optical networks.

A BER test set is used to generate the bit sequence with a reference bit rate. The attached subsystem in the model is a small system, which allows data sequence to be converted into I and Q symbols using QAM sequence generator. The FFT (Fast Fourier Transform) points in OFDM modulator should always be twice as much as the number of subcarriers at the very least [13].

As shown in the figure, 4-QAM OFDM is being used and laser is being passed through free space to reach the receiver side. It can also be seen that before laser reaches the channel, it is first passed into the MZ modulator. MZ modulator stands for Mach–Zehnder modulator and its role is to control the amplitude of the optical wave.

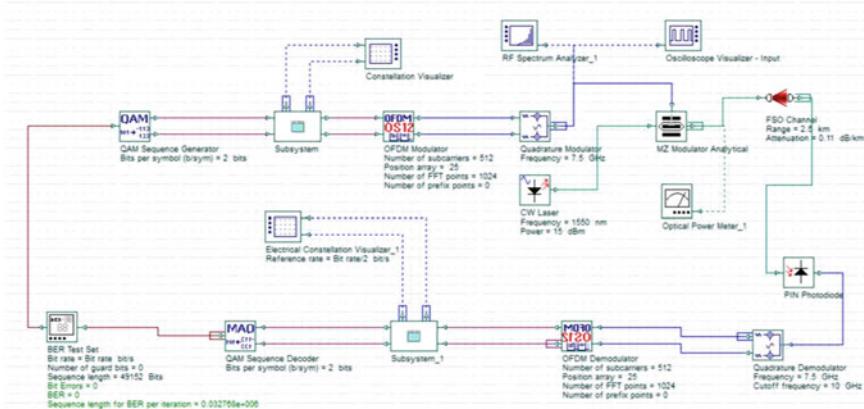


Fig. 3 Schematic

Table 1 FSO channel parameters

FSO channel parameters	Values
Attenuation constant	0.11 dB/km
Beam divergence	2 mrad
Transmitter aperture diameter	5 cm
Receiver aperture diameter	20 cm

Table 2 Layout parameters

Layout parameters	Values
Bit rate	15–20 Gbps
Sequence length	32,768
Samples per bit	5
Number of samples	131,702

This project uses a CW laser as the light source, which is a continuous wave laser. It is a beam of laser with its heat output controlled to a certain degree. This light source passes through an FSO channel, which is basically free space. To receive the signal after it passes through the free space channel, a photodiode is used. Photodiodes are devices that are used as receivers in optical communication because they can convert light to electric current. The most common type of photodetector used is “Avalanche Photodetectors” [14]. It is then passed through the receiver side to obtain the result.

The constellation diagram visualizer that is connected is used to show the representation of the QAM modulated signal of XY plane. It allows us to see the ideal vs actual plot that we got.

The purpose of RF spectrum analyzer is to display a signal in frequency domain where RF stands for radiofrequency. The oscilloscope visualizer shows the amplitude of the signal against the time sweep.

The specifications of all the components are shown in Tables 1, 2, 3 and 4.

Table 3 Laser parameters

Laser parameters	Values
Frequency	1550 nm
Power	25 dB

Table 4 OFDM modulator parameters

OFDM modulator parameters	Values
Number of input ports	1
Number of subcarriers	512
Number of prefix points	0
Number of FFT points	1024
Position array	25

4 Result and Discussion

In this section, we simulate the system and analyze the results. This paper studies the values of BER by varying several parameters. Bit error rate values are plotted and tabulated for different weather conditions. It is also seen how BER varies with power. Finally, it is analyzed how BER changes with the distance at different data rates.

The graph shown in Fig. 4 is of $\text{BER} \times \text{Range}$ with varying power. BER indicates the number of bit errors. Thus, less BER means better data transmission. The BER values at different amounts of power were observed.

Upon analyzing, we saw that with an increase in power, the BER decreased. Thus, we could conclude more amount of power reduces the amount of data loss.

Since, we got the least BER when the light source operated at a power of 25 dB, the rest of the plots in this work were taken at the same amount of power.

BER values were observed for three different weather conditions, which were clear weather, hazy and fog. The attenuation constants for these weather conditions are 0.11, 4.2 and 25.5 dB/km, respectively [15].

From the results shown in Table 5, we can conclude that as the weather conditions get worse, we get high BER values at shorter distances. This work only focuses on clear weather conditions, therefore the attenuation constant taken for the rest of the plots is 0.11 dB/km.

Next, $\text{BER} \times \text{Link Range}$ graph was taken for very high data variations and it was observed that as the data rate increased, BER increased drastically. The BER variation from 10 to 50 Gbps can be seen in Fig. 5.

Since such high BER values were obtained, smaller data rates were chosen, which were more suitable to the model. Figure 6 shows the plot of $\text{BER} \times \text{Range}$ for less data rates. As shown in the graph, we obtain much better BER values at these data

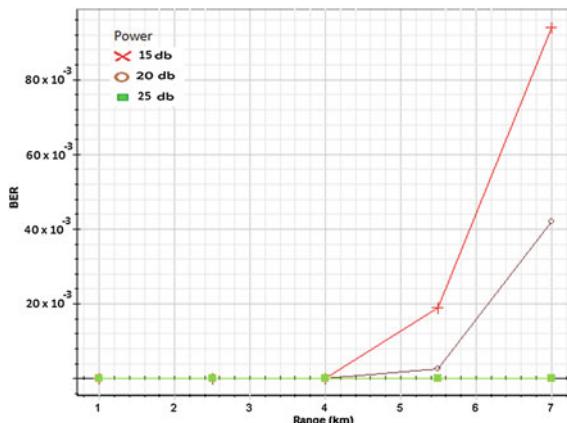
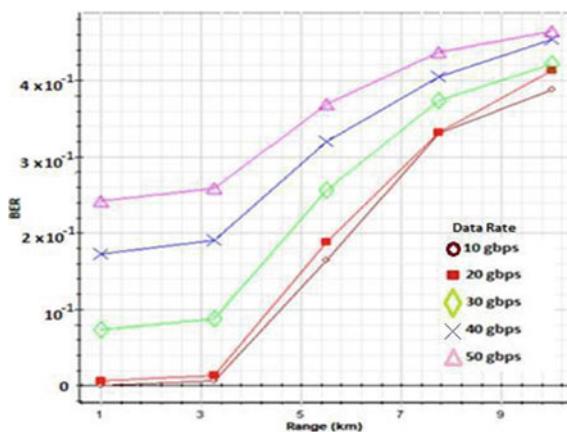


Fig. 4 BER versus range with varying power

Table 5 BER values at different weather conditions

Attenuation constant	Weather conditions	BER
0.11 dB/km	Clear weather	0 at 7.60 km
		29.60e-006 at 7.65 km
		11.43e-005 at 7.70 km
4.20 dB/km	Haze	0 at 2.50 km
		61.03e-006 at 2.55 km
		91.55e-005 at 2.60 km
25.5 dB/km	Fog	0 at 0.78 km
		30.51e-006 at 0.79 km
		12.20e-005 at 0.80 km

**Fig. 5** BER versus range for high data rates

rates—15, 17.5, and 20 Gbps. The BER values for these data rates can be seen and compared from Table 6.

We also obtain RF spectrum, quadrature modulator output at the transmitter side, transmitted QAM constellation and received QAM constellation shown in Figs. 7, 8, 9 and 10.

Fig. 6 BER versus range with varying data rates

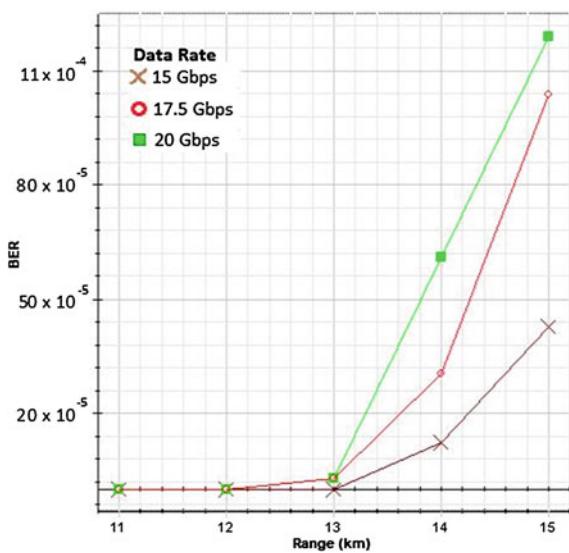


Table 6 BER at varying data rates and link ranges

Data rate	Range = 11 km	Range = 12 km	Range = 13 km	Range = 14 km	Range = 15 km
15 Gbps	BER = 0	BER = 0	BER = 0	BER = 12.2e-05	BER = 42.7e-05
17.5 Gbps	BER = 0	BER = 0	BER = 30.1e-06	BER = 30.5e-05	BER = 10.3e-04
20 Gbps	BER = 0	BER = 0	BER = 30.5e-06	BER = 61.0e-06	BER = 11.9e-04

Fig. 7 Quadrature modulator output

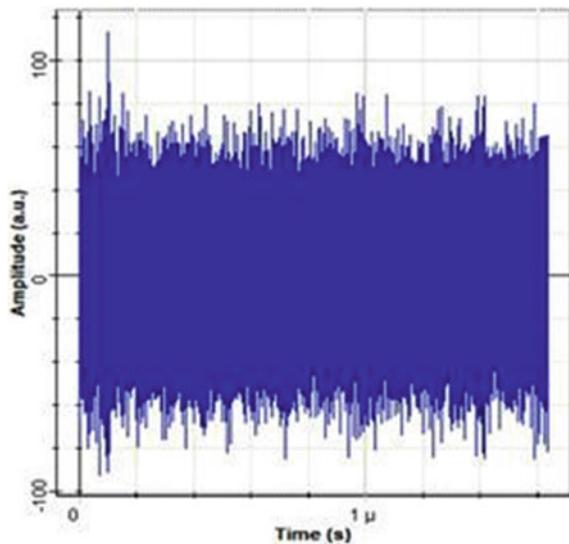


Fig. 8 RF spectrum at transmitter

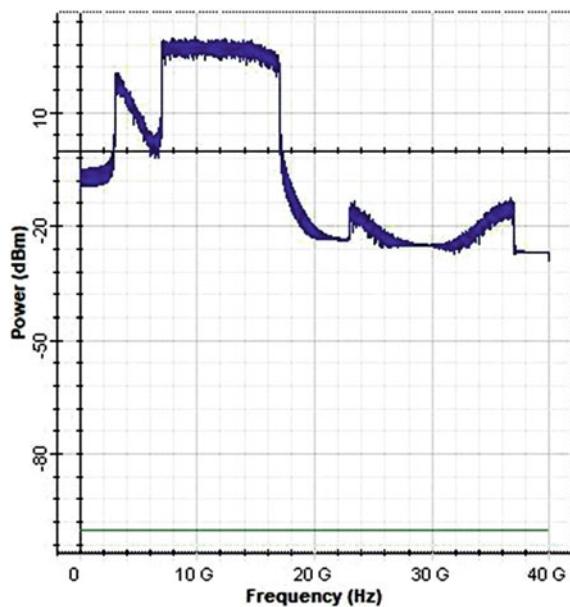


Fig. 9 Transmitted QAM constellation

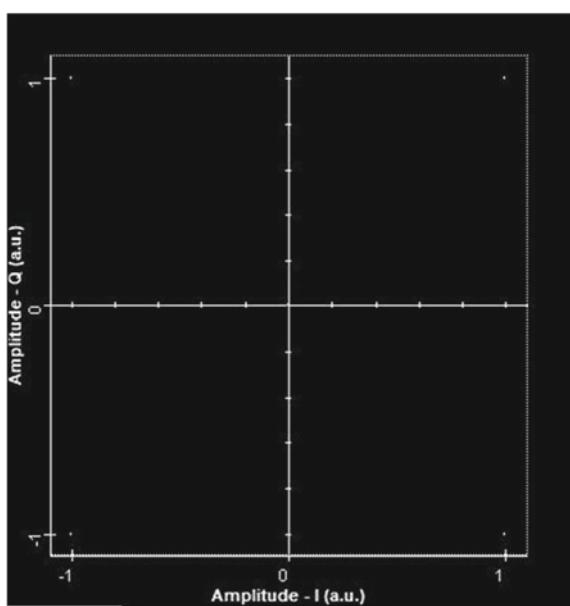
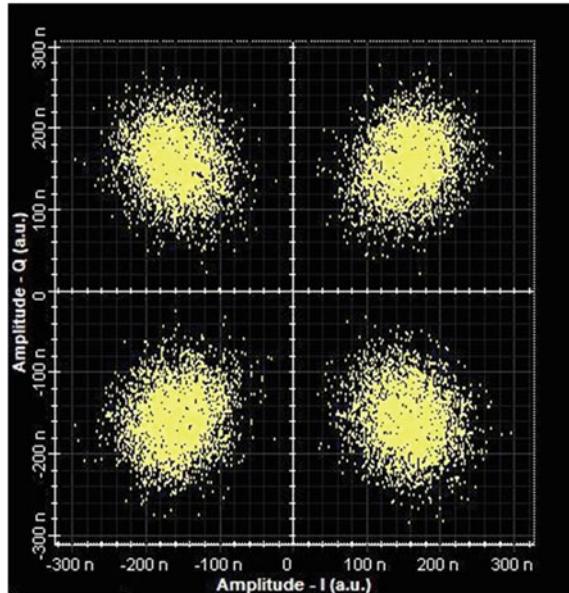


Fig. 10 Received QAM constellation



5 Conclusion

In this work, we have analyzed the system performance of FSO-OFDM system for different bit rates and link ranges. From the results obtained in Table 6, we conclude that we get the least BER at a range of 13 km at a data rate of 15 Gbps when the light source is operating at 25 dB power. BER values observed beyond that range are not permissible. Therefore, the proposed system supports data rates of up to 15 Gbps till 13 km range for clear weather conditions.

OFDM can be used to transmit a large amount of data with minimum interference. With its performance improving over the years, it could become a very efficient technology for communication systems in the future.

References

1. H. Kaur, K. Singh, T. Singh, A review on OFDM-FSO system. *Adv. Res. Electr. Electron. Eng.* **5**(2), 110–112 (2018)
2. A. Malik, P. Singh, Free space optics: current applications and future challenges. *Int. J. Opt.* (2015)
3. R. Gupta, T. Kamal, P. Singh, Performance of OFDM: FSO communication system with hybrid channel codes during weak turbulence. *J. Comput. Netw. Commun.* (2019)
4. S. Bloom, E. Korevaar, J. Schuster, H. Willebrand, Understanding the performance of free-space optics. *J. Opt. Netw.* (2003)
5. P. Vohra, A. Sharma, S. Kaur, M. Kaur, A. Kesarwani, Analysis of performance of FSO link during the months of rainfall in Bengaluru, India, in *2018 2nd International Conference on*

- Micro-electronics and Telecommunication Engineering (ICMETE)*, Ghaziabad, India (2018), pp. 12–14. <https://doi.org/10.1109/ICMETE.2018.00016>
- 6. S. Kaur, A. Kakati, Analysis of free space optics link performance considering the effect of different weather conditions and modulation formats for terrestrial communication. *J. Opt. Commun.* (2018)
 - 7. S. Kaur, Performance analysis of FSO link under the effect of fog in Delhi region, India. *J. Opt. Commun.* (2020)
 - 8. S. Weinstein, The history of orthogonal frequency-division multiplexing [History of Communications]. *IEEE Commun. Mag.* **47**(11) (2009)
 - 9. D. Khosla, OFDM modulation technique & its applications: a review. *Int. Conf. Innov. Comput.*, 101–105 (2017)
 - 10. A. Ali, A. Barakatze, Behavior and techniques for improving performance of OFDM systems for wireless communications. *Int. J. Adv. Res. Comput. Eng.* **4**(1) (2015)
 - 11. S. Bawazir, P. Sofotasios, S. Muhaidat, Y. Al-Hammadi, G. Karagiannidis, Multiple access for visible light communications: research challenges and future trends. *IEEE Access* **6**, 26167–26174 (2018)
 - 12. H. Li, A. Abdi, O. Somekh, OFDM Modulation classification and parameters extraction, cognitive radio oriented wireless networks and communications, (2006)
 - 13. A. Hamza, J. Deogun, D. Alexander, Classification framework for free space optical communication links and systems. *IEEE Commun. Surv. Tutor.* **21**(2) (2019)
 - 14. S. Selvendran, A. Raja, K. Muthu, A. Lakshmi, certain investigation on visible light communication with OFDM modulated white LED using optisystem simulation. *Wirel. Pers. Commun.* (2019)
 - 15. N. Mohammed, A. Al-Wakeel, A.M. Aly, Performance evaluation of FSO link under NRZ-RZ line codes, different weather conditions and receiver types in the presence of pointing errors. *Open Electr. Electron. Eng. J.* (2012)

Customer Churn Prediction in Telecommunication Using Gradient Boosting Machine



Manoj Kumar and Dharmendra Kumar Yadav

Abstract Machine Learning is being used extensively to solve problems in various fields. Predictive analytics is one of the major applications of machine learning. Predictive analytics uses many techniques, i.e., machine learning, data mining, artificial intelligence, and statistics. It can use present data to make future predictions. Churn prediction is one of the areas where predictive analytics can be applied. The word ‘churn’ can be used in many contexts with different meanings, but in this paper, it refers to a situation when a customer stops using any company’s products or services. Customer Churn Prediction (CCP) is important because the profitability of any company is directly affected by it. It is much costlier to find a new customer than to keep an existing customer who is loyal to the company. If churning customers can be predicted in advance, the company can approach and retain them using some retention mechanism. This paper proposes a churn prediction model that uses the Gradient Boosting Machine (GBM) classifier to identify the customers who may churn. After the classification, the performance of the customer is evaluated using various performance metrics. Accuracy is not a significant metric in performance evaluation as the churn dataset is imbalanced, leading to a misleading result. To overcome this problem, we have considered two more metrics: PR plot and ROC curve.

1 Introduction

A large volume of data is being generated at a fast rate by telecom companies [13]. Every telecom company competes to improve its client share. Customers may have options for better and cost-effective products and services. The goal is to continue and be profitable in this competitive environment. Customer churn occurs when many

M. Kumar (✉) · D. Kumar Yadav
MNNIT Allahabad, Department of Computer Science, Allahabad, India
e-mail: manojk@mnnit.ac.in

D. Kumar Yadav
e-mail: dky@mnnit.ac.in

customers are not satisfied with the product or services of the company. Customer churn may occur in many areas like retail, telecommunication, insurance, banking, credit cards, gaming, or many types of subscription services. Customers are never bound to use the companies' services in non-contractual settings, i.e., prepaid customers in telecom.

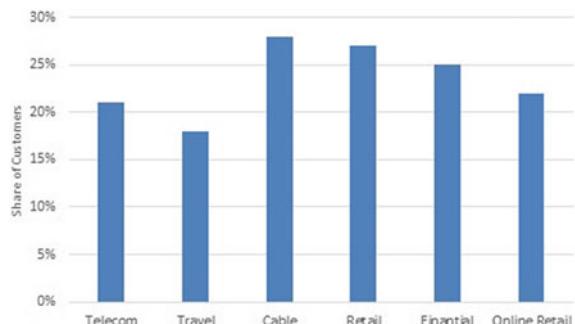
Profit maximization is the goal of any company. Profit decreases with the increase in the churn rate. Churn rate increases when a large number of the customer base is unsatisfied with its services or products of a company. There could be many reasons why a customer churns. But as per the telecom perspective, some of the main reasons are Call Facilities, Network Quality, Internet Facilities, Alerts, Booster Facilities, Customer Service, etc.

Attracting new customers is costlier than retaining existing customers [2]. The cost of attracting new customers is almost five to six times more expensive than keeping the existing ones. Hence, customer churn prediction is essential. Customer retention can overturn the company's situation. Hence, a churn prediction model followed by a retention model is the utmost requirement of any company. Churn can affect the revenue growth of a company adversely [8].

The skewed nature of churn data makes the classification difficult as there is a large difference in the number of instances of different classes. Hugeness and sparsity also create significant difficulties for customer churn. It is a challenge to preprocess a large volume of data. To handle this issue, a bundle of mechanisms is provided in the literature. Preprocessing is done to reduce the high dimensionality of data. Churn prediction is a binary classification problem with two classes, churner and non-churner. These problems possess difficulties dealing with the class imbalance problem. It involves a huge cost to the company for misclassifying churners as non-churners since the chunner class is of primary interest. Often, the majority class biases the classifier toward itself, and finally, a classifier classifies entirely as a majority class ignoring the minority class. Figure 1 shows the churn rate in the United States of America.

The organization of the paper is as follows. The literature survey has been presented in Sect. 2. In Sect. 3, the proposed model is described. Section 4 discusses

Fig. 1 Churn rate of USA in 2018 industry-wise



the various performance metrics relevant to churn prediction. Section 5 discusses the experimentation and dataset requirement. The last section is about the conclusion and future scope.

2 Related Work

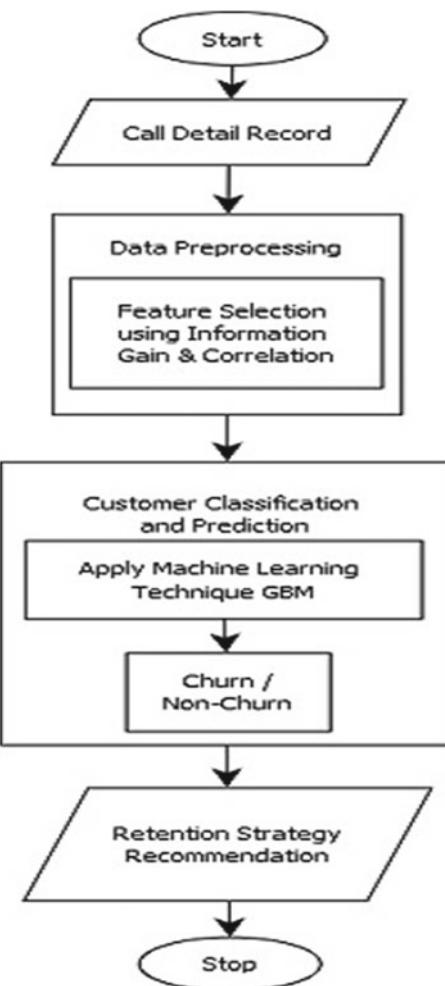
Churn prediction has been implemented through various techniques such as Data Mining, Machine Learning, Meta-heuristic approach, and Probabilistic approach. These techniques help companies to find out and retain churn customers. The Decision Tree is mostly used for churn prediction in the literature due to its interpretation capabilities. It performs better for linearly related data.

Ullah et al. [2] proposed a churn prediction model that applied clustering as well as classification techniques in the same model to find out the churn customers and factors responsible for churning. It uses Random Forest for classification and K-Means for customer segmentation. The model has shown an accuracy of 88.63 percent. Vijaya et al. [3] proposed an ensemble technique. It uses fuzzy-based clustering techniques, namely Possibility C-Means (PCM), Fuzzy C-Means (FCM), and Possibility Fuzzy C-Means (PFCM), to segment the customers into groups. These groups are further partitioned into training and testing data using the holdout method. In the last step, various ensemble methods like Boosting, Bagging, and Random Space are applied for building the model. Amin et al. [6] proposed a novel technique that divides the dataset into two zones based on the distance, which is further divided into two groups: (i) data with low certainty and (ii) data with high certainty. It shows that there is a strong correlation between the distance factor and the certainty of the classifier. Vijaya et al. [12] proposed a model that involves Particle Swarm Optimization (PSO). It proposed three variants of PSO, namely PSO-FS, PSO-SA, and PSO-FSSA. Amin A. et al. [13] proposed a rule-based technique based on Rough Set Theory (RST) to extract key decision rules for customer churn and non-churn. Kim et al. [14] proposed a new technique for churn prediction that involves communication patterns among telecom customers and propagating churning information from chunner to non-chunner in their network connection.

3 Proposed Model

Figure 2 shows the proposed CCP model. Data preprocessing is performed in the first step. Irrelevant features are eliminated using the Information Gain and Correlation Attribute Ranking Filter technique of the Weka toolkit. In the second step, the GBM classification algorithm is applied over the dataset to identify churn customers. The model recommends a retention strategy for churn customers in the final step.

Fig. 2 Proposed model for customer churn prediction



4 Performance Metrics

There are several performance metrics used for the evaluation of various classifiers and clustering models. However, some of the most important metrics used in churn prediction models are Precision-Recall plot, F-measure, AUROC curve, Misclassification cost (Type-I, Type-II), Probability of Detection (POD), Probability of False Alarm (POF), Decile-lift, etc.

The problem with accuracy is that it works better for a balanced class distribution of the dataset, i.e., the number of instances of various classes is almost equal. Otherwise, it would give a misleading result.

Table 1 Confusion matrix

	Class0 (Predicted)	Class1 (Predicted)
Class0 (Actual)	True-Positive (TP)	False-Negative (FN)
Class1 (Actual)	False-Positive (FP)	True-Negative (TN)

Confusion Matrix: A confusion matrix is normally used to investigate further the performance of a classifier over the dataset given as shown in Table 1.

Precision (P): It is the number of positive predicted instances belonging to the positive class. The data points predicted as positive are truly positive [19], as shown in Eq. (1).

$$P = \frac{TP}{TP + FP} \quad (1)$$

Recall (R) (Sensitivity or POD): It is the number of positive predicted instances out of the total number of positive instances in the dataset, as shown in [19] and in Eq. (2).

$$R = \frac{TP}{TP + FN} \quad (2)$$

Accuracy (A): It is the percentage of correctly classified instances as churning or non-churning as shown in Eq. (3). Accuracy assumes equal misclassification cost for False Positive and False Negative that is not suitable for imbalanced class distribution. An accuracy of 99 percent can be excellent or poor.

$$A = \frac{TP + FN}{TP + TN + FP + FN} \quad (3)$$

Area Under the Curve (AUC): One of the most important metrics for the evaluation of the performance of a classifier is AUC. It is also described as Area Under the Receiver Operating Characteristics (AUROC). AUC measures the degree of separability, and ROC is a probability curve. A higher value model is treated as a better model to distinguish between non-churn and churn customers.

Precision-Recall plot (PR plot): It is a handy plot between precision and recall. It is important when the class distribution is not balanced. This is because the PR plot does not use True Negatives like the AUC curve.

5 Experimentation and Performance Evaluation

Dataset Description: A standard telecom dataset (Telco Customer Churn) is taken from the KDD library for experimentation that contains 21 attributes. It has 5000 observations. Fourteen attributes remain after the feature selection step using the

Fig. 3 Accuracy comparison of ML algorithms



Information Gain Correlation Attribute of the Weka tool. All the coding has been performed, and evaluations are done in Python 3.7. In our experiment, we have taken into consideration three performance metrics, namely PR plot, Accuracy, and AUC curve, for evaluation purposes.

Gradient Boosting Machine (GBM) was applied to the dataset after Weka's feature selection using the attribute ranking method. Those attributes which are not important to the model building are eliminated by this method. We trained the model using GBM in two ways (Figs. 3 and 4):

- (i) Dataset without cross-validation;
- (ii) Dataset with tenfold cross-validation to avoid any bias.

It is found that GBM shows an accuracy of 88.4% with a dataset without cross-validation. In the case of tenfold cross-validation, 81.5% accuracy is recorded.

We have compared the accuracy of GBM to three standard benchmark machine learning algorithms that are Support Vector Machine (SVM), Decision Tree (DT), and Artificial Neural network (ANN) and found the following result, as shown in Fig. 3. The result shows that the GBM machine learning algorithm is a clear winner.

The second metric used for evaluating the model is the AUC curve. The above-mentioned four machine learning algorithms (GBM, SVM, DT, and ANN) were applied to the dataset, and the AUC value was calculated as shown in Fig. 4(a). The result shows that GBM performs better than the other three.

The third metric used for evaluating the model is the PR plot. Again the four machine learning algorithms (GBM, SVM, DT, and ANN) were applied to the dataset, and the PR value was calculated as shown in Fig. 4(b). The result shows that GBM performs better than the other three.

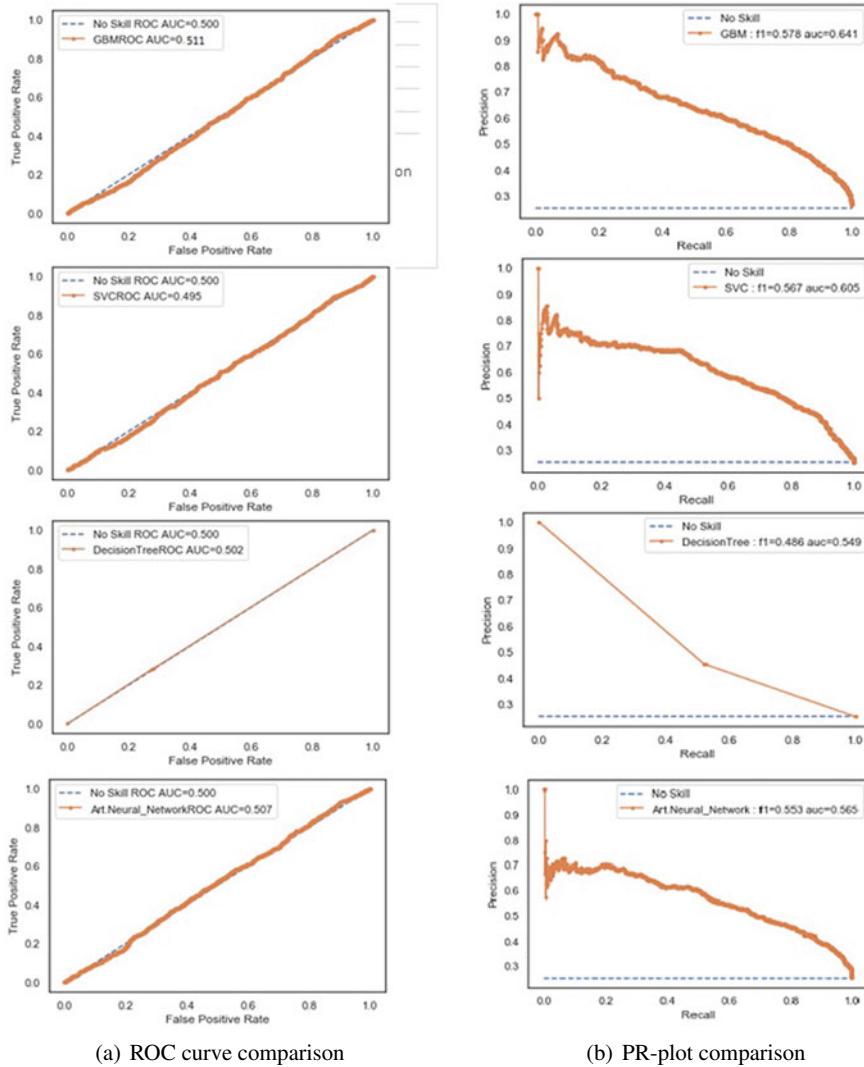


Fig. 4 Performance Comparison

6 Conclusion and Future Scope

Customer churn prediction is an important issue of CRM used to forecast client share of any company and help in building a retention model to preserve churning customers. It is important to build a model that is reliable and useful in recommending a retention strategy. Retention strategy could include services/offers to the potential churn customers through email or messages. In our study, a CCP model is

trained/presented that uses a GBM classifier. The model is validated through three standard evaluation metrics, namely Accuracy, PR plot, and AUC curve. GBM gives an excellent result with an accuracy of 88.4 percent, which is better than many approaches presented in the literature and can further be improved. In the future, we can apply artificial intelligence, the meta-heuristic approaches, or the probabilistic approaches to the problem at hand. We can also consider eliminating imbalance in the churn dataset before or during the model building to avoid any biases.

References

1. J. Han, M. Kamber, Data mining: concepts and techniques. Morgan Kaufmann (Elsevier) (2006)
2. I. Ullah, B. Raza, A.K. Malik, M. Imran, A.U. Islam, S.W. Kim, A churn prediction model using random forest: analysis of machine learning techniques for churn prediction and factor identification in telecom sector. *IEEE Access* **7**, 60134–60149 (2019)
3. J. Vijaya, E. Sivasankar, S. Gayathri, Fuzzy Clustering with Ensemble Classification Techniques to Improve the Customer Churn Prediction in Telecommunication Sector, in *Recent Developments in Machine Learning and Data Analytics* (Springer, Singapore, 2019), pp. 261–274
4. A. Amin, B. Shah, M.K. Khattak, Cross-company customer churn prediction in telecommunication: A comparison of data transformation methods. *Int. J. Inf. Manage.* **46**, 304–319 (2019)
5. H. Ali, M.N.B. MohdSalleh, K. Hussain, M.F. Mustaq, Imbalance class problems in data mining: a review. *Indones. J. Electr. Eng. Comput. Sci.* **14**(3), 1560–1571 (2019)
6. A. Amin, F. Al-Obeidat, B. Shah, A. Adnan, Customer churn prediction in telecommunication industry using data certainty. *J. Bus. Res.* **94**, 290–301 (2019)
7. A.D. Caigny, K. Coussement, K.W.D. Bock, A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *Eur. J. Oper. Res.* **269**(2), 760–772 (2018)
8. Mishra and U. S. Reddy, A novel decision tree based on profit variance maximization criterion for customer churn prediction, in *10th International Conference on Intelligent Human-Machine Systems and Cybernetics* (2018)
9. X. Zhang, Z. Zhang, D. Liang, H. Jin, A novel decision tree based on profit variance maximization criterion for customer churn problem, in *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol 1 (IEEE, 2018)
10. A. Amin, B. Shah, A. M. Khattak, T. Baker, H. ur Rahman Durani, S. Anwar, Just-in-time customer churn prediction: With and without datatransformation, in *Proc. IEEE Congr. Evol. Comput., Jul. 2018*, pp. 1–6
11. S. Hopner, E. Stripling, B. Baesens, S.V. Broucke, T. Verdonck, Profit driven decision trees for churn prediction. *Eur. J. Oper. Res.* **269**(2), 760–772 (2018)
12. J. Vijaya, E. Sivasankar, An efficient system for customer churn prediction through particle swarm optimization based feature selection model with simulated annealing. *Clust. Comput.*, 1–12 (2017)
13. A. Amin, S. Anwar, M. Nawaz, Customer churn prediction in the telecommunication sector using a rough set approach. *Neurocomputing* **237**, 242–254 (2017)
14. K. Kim, C. Jun, J. Lee, Improved churn prediction in telecommunication industry by analyzing a large network, PAKDD 2011. *Expert. Syst. Appl.*, 411–422 (2014)

Blood-Based DNA Methylation Marker Identification for Parkinson's Disease Prediction



Jisha Augustine and A. S. Jereesh

Abstract Parkinson's disease (PD) is one of the most common age-related neurodegenerative diseases. Identifying PD using a minimally invasive method improves diagnosis, prognosis and treatment selection. Using machine learning analysis of genome-wide DNA methylation profiles, we investigated CpG methylation status's utility to differentiate blood from PD patients to a healthy one. This paper proposes a two-level feature selection method with Fast Correlation-Based Filter (FCBF) in the first level and Recursive Feature Elimination with Logistic regression classifier (LR-RFE) in the next level to identify significant CpG sites. A stacked ensemble classifier with Naive Bayes (NB), k-nearest neighbors (kNN) and Linear Discriminant Analysis (LDA) as base models and Random forest (RF) as meta-model showed the highest classification performance. The area under the ROC Curve (AUC), sensitivity and specificity have been used for evaluation purpose. Selected 29 CpG sites could successfully classify PD with 92% AUC in 10-fold cross-validation with repeats 5. Test with hidden samples proved the relevance of the selected CpG sites.

Keywords Parkinsons disease · DNA methylation · Machine learning · Two-level feature selection · Classification · Stacked ensemble

1 Introduction

Parkinsons disease (PD) is one of the most common neurodegenerative diseases that affect the elderly population. Loss of neurons in the specific areas of substantia nigra in the mid-brain and accumulation of Lewy bodies are the primary characteristics of PD. Affected individuals suffer from motor and non-motor symptoms. The genetic,

J. Augustine (✉) · A. S. Jereesh

Bioinformatics Lab, Department of Computer Science, Cochin University of Science and Technology, Kochi 682022, Kerala, India

e-mail: jishaugustine@cusat.ac.in

A. S. Jereesh

e-mail: jereesh@cusat.ac.in

environmental and lifestyle factors cause PD [1]. As PD symptoms appear at a later stage of the disease, identification of biomarkers facilitates diagnosis, prognosis and treatment selection. Blood biomarkers are minimally invasive, cheap and easily accessible [2].

DNA methylation is the biological process which affects the functioning of gene and gene expression without changing the DNA sequence. In this process, a methyl group is added to DNA generally to the fifth position of cytosine. In mammals, most of the DNA methylation occurs at cytosine followed by a guanine called CpG sites [3]. Increasing evidence shows that environmental and lifestyle factors may trigger alterations of DNA methylation [4, 5]. These factors made researchers focus on studying the influence of DNA methylation on PD. Large scale epigenome-wide association study provides plentiful data to analyse the influence of DNA methylation on different diseases [6, 7]. Machine learning has achieved a significant role in the analysis of such large scale data [8].

In this study, we focus on identifying marker CpG sites by analysing DNA methylation profiles from blood samples for PD prediction. We used two-level feature selection for marker sites identification and stacked ensemble learning for classification.

This article is organised into five sections. Section 2 discusses the existing works related to this paper. The details of dataset and the proposed methodology is described in Sect. 3. Section 4 explains and analyses the results followed by the conclusion of the paper in Sect. 5.

2 Related Work

Early studies in brain centred on DNA methylation-dependent performance on SNCA [9]. Later, the association of DNA methylation on the regulation of several PD specific genes and the role of DNA methylation in the development of PD were observed in various studies [10, 11].

A study of DNA methylation in brain and blood samples of PD patients identified distinctive methylation patterns involving different genes and suggested blood as a surrogate for brain tissue in the DNA methylation analysis of PD [12]. An Epigenome-wide association study using 335 PD and 237 control blood DNA samples conducted by Chuang et al. [13] generated DNA methylation data and identified 82 significant CpGs sites. A comparative study of different feature selection algorithms on case-control DNA methylation data with 43 controls and 23 diseased PD samples identified significant transcripts using Logistic Regression (LR) and Random Forest (RF) [14]. Wang et al. [15] integrated Gene expression and DNA methylation data of PD blood samples and identified 53 hypo-methylated upregulated gene signature as PD biomarkers using RF. A recent study did a meta-analysis on 229 K CpG probes in 1,132 cases and, 999 controls from two independent blood-based DNA methylation cohorts of PD identified two previously unreported associations between DNA methylation and PD [16].

From the existing works, it has been observed that blood-based DNA methylation analysis has a vital role for the identification of PD and feature selection methods could select significant CpG sites, which effectively classify PD from these large scale data.

3 Materials and Methods

Genome-wide DNA methylation data (GSE111629) [13] for this study is collected from Gene Expression Omnibus (GEO) [17]. The dataset contains data from 335 PD patients and 237 healthy controls in whole blood samples. The data was measured by the Illumina Infinium 450k Human Methylation Beadchip. We have downloaded the background normalised methylation data (β -values) with 486,000 CpGs sites. Missing values were handled by knn imputation [18] and probes mapped to sex chromosomes were removed to prevent bias due to gender effects [19]. M-value is calculated from the β -values of preprocessed and filtered data for differential methylation analysis [20]. Differentially methylated sites were identified using moderated t-statistics using Limma package [21] with Benjamini-Hochberg FDR < 0.05.

The samples in the resulting dataset were randomly split into two cohorts: 80% samples for feature selection, training and cross-validation, the remaining 20% hidden samples for testing. We have performed a two-level feature selection. In the first level, the relevant features are selected using Fast Correlation-based Filter (FCBF) [22], a fast filter method, which uses symmetrical uncertainty-based correlation to select relevant features. In the next level, a wrapper method Recursive Feature Elimination [23] with logistic regression classifier (LR-RFE) to select optimum features.

The data with selected features underwent a classification process. The stacked ensemble learning algorithm [24] passes predictions from base models to the meta-model for selecting an optimum combination of the predictions. We used Naive Bayes (NB), k-nearest neighbors (kNN) and Linear Discriminant Analysis (LDA) as base models and Random Forest (RF) as the meta-model (Stacked-RF). The workflow of the proposed methodology is given in Fig. 1. We performed repeated tenfold cross-validation with repeats 5. The performance of the learning algorithm is assessed using AUC, specificity and sensitivity.

4 Results and Discussion

From 486,000 CpG sites, we have identified 16,642 differentially methylated sites after filtering and differential expression analysis.

The dataset with selected CpG sites underwent two-level feature selection. The FCBF feature selection identified 41 features (CpG sites) as the relevant subset. For selecting an optimal feature set, we have performed wrapper based LR-RFE feature selection. As a result, 29 features with 18 hypo-methylated and 11 hyper-methylated

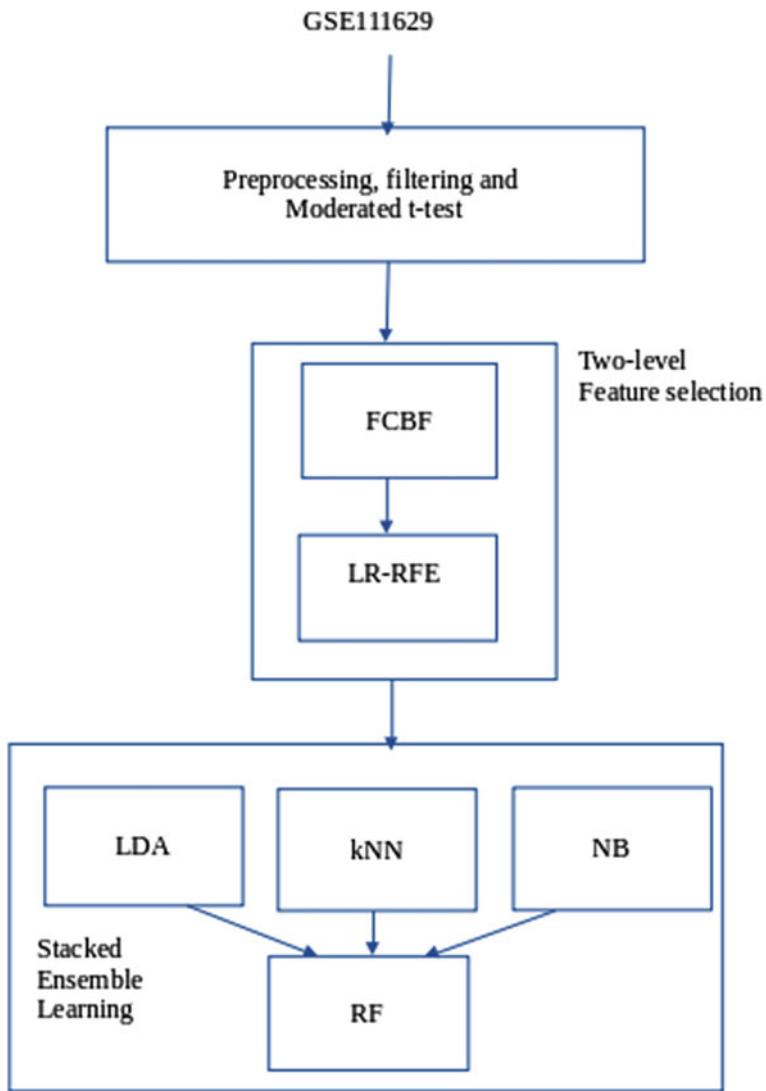


Fig. 1 Workflow of the proposed method for optimum feature selection and classification

sites were identified as the optimum feature set. Twenty selected sites belong to 20 unique genes, and remaining are intergenic CpGs. Table 1 shows the selected CpG sites and associated genes.

cg01316378, cg08066645, cg09552548, cg10433043, cg11229399, cg14402574, cg26911986 and cg27655512 are hypo-methylated intergenic sites and cg26645509 is hyper-methylated intergenic site.

Table 1 Selected CpGs and associated genes with gene region and direction of methylation

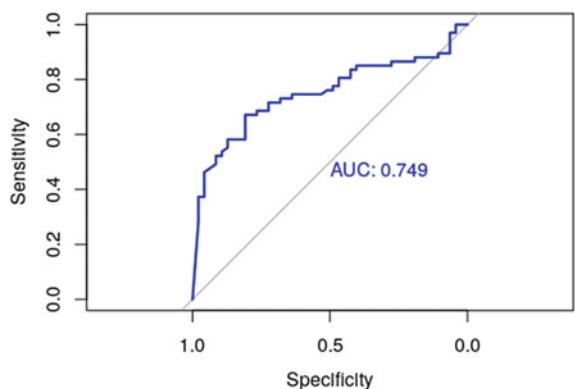
CpG sites	Gene region	Gene symbol	Direction of methylation
cg04368939	Body	KCNIP2	Hypo-methylated
cg02845997	Body	ELK3	Hypo-methylated
cg17107156	Body	CHRNBI	Hypo-methylated
cg14118546	TSS1500	KIAA1012	Hypo-methylated
cg05001044	TSS1500	MIR1977	Hypo-methylated
cg04731861	Body	ARPC2	Hypo-methylated
cg17976473	TSS1500	DMBT1	Hypo-methylated
cg25782229	Body	WT1	Hypo-methylated
cg04699460	Body	THUMPD1	Hypo-methylated
cg00264578	Body	CDH4	Hypo-methylated
cg07753891	TSS200	ELMOD2	Hypo-methylated
cg25265769	Body	CTNNA3	Hypo-methylated
cg14204081	Body	C6orf208	Hypo-methylated
cg02771117	3'UTR	FAM167A	Hypo-methylated
cg25193885	Body	SHANK2	Hypo-methylated
cg09163005	TSS1500	H2AFJ	Hypo-methylated
cg18199664	TSS200	NRXN3	Hypo-methylated
cg18476566	1stExon	OR4K14	Hypo-methylated
cg14488391	Body	OAZ2	Hypo-methylated
cg08843279	TSS1500	NRIP1	Hypo-methylated

To assess the performance of the selected features, the dataset with optimum features were given to stacked ensemble classifier. We have compared the performance of our proposed feature selection method with FCBF and LASSO using different machine learning algorithms. Table 2 shows result in terms of AUC, specificity and sensitivity on different classifiers.

The proposed two-level feature selection method uses a filter method FCBF in the first level and a wrapper method LR-RFE in the second level. Filter methods are fast, but the result is not always optimum, and wrapper methods select more optimal features but time-consuming [25]. Applying the wrapper method in the second level helps to select a minimum number of optimum features with less time. As a result, 29 features are selected. From the comparative study, all classifiers except kNN showed highest AUC, specificity and sensitivity with the proposed feature selection method. LASSO selected 373 features and showed less AUC in all classifiers. FCBF selected 41 features showed reasonable performance. In both FCBF and the proposed method, Stacked ensemble with RF as meta learner showed the highest performance. Even though both had similar performance, the proposed method selected optimum subsets from the 41 features selected by FCBF.

Table 2 AUC, specificity and sensitivity of different feature selection methods

Classifier	Feature selection	No. of features	AUC(%)	Specificity(%)	Sensitivity(%)
NB	LASSO	373	88	87.9	74.3
kNN			69	65.8	69
LDA			56	85.8	28.4
RF			72	61.6	77.6
Stacked-RF			70	74.8	52.6
NB	FCBF	41	85	81.1	76.5
kNN			77	67.4	79.1
LDA			88	82.1	78.4
RF			87	77.9	82.8
Stacked-RF			92	87.0	80.8
NB	FCBF+LR-RFE	29	88	80.0	84.3
kNN			75	72.1	70.1
LDA			89	82.6	79.9
RF			89	82.1	84.2
Stacked-RF			92	87.1	80.2

Fig. 2 Performance of Stacked-RF with hidden dataset

To validate the reliability of selected CpG sites and performance of the proposed model, we have tested with the hidden dataset. 74.9% AUC proved the reliability of the selected features (Fig. 2).

5 Conclusion

DNA methylation plays an important role in PD development. In this study, we analysed DNA methylation profiles of PD from blood samples. We identified 29 CpG sites which effectively distinguishes PD patients from healthy one using two-level feature selection and stacked ensemble classification. Test using hidden data proved the reliability of selected sites. Further biological validations and clinical trials are required to evaluate the clinical significance of these selected markers. Moreover, we believe that our study can help to provide a minimally invasive method for PD diagnosis and prognosis. Future work includes the generation of biologically more relevant sites by incorporating pathway information. Adding more data will make the prediction more effective.

References

1. R. Balestrino, A.H. Schapira, Parkinson disease. *Eur. J. Neurol.* **27**(1), 27–42 (2020)
2. M. Thambisetty, S. Lovestone, Blood-based biomarkers of Alzheimer's disease: challenging but feasible. *Biomark. Med.* **4**(1), 65–79 (2010)
3. B. Jin, Y. Li, K.D. Robertson, DNA methylation: superior or subordinate in the epigenetic hierarchy. *Genes Cancer* **2**(6), 607–617 (2011)
4. Z. He, R. Zhang, F. Jiang, W. Hou, C. Hu, Role of genetic and environmental factors in DNA methylation of lipid metabolism. *Genes Dis.* **5**(1), 9–15 (2018)
5. E.M. Martin, R.C. Fry, Environmental influences on the epigenome: exposure-associated DNA methylation in human populations. *Annu. Rev. Public Health* **39**, 309–333 (2018)
6. J.M. Flanagan, Epigenome-wide association studies (EWAS): past, present, and future, in *Cancer Epigenetics* (Humana Press, New York, NY, 2015), pp. 51–63
7. M.A. Mooney, P. Ryabinin, B. Wilmot, P. Bhatt, J. Mill, J.T. Nigg, Large epigenome-wide association study of childhood ADHD identifies peripheral DNA methylation associated with disease and polygenic risk burden. *Transl. Psychiatry* **10**(1), 1–12 (2020)
8. S. Rauschert, K. Raubenheimer, P.E. Melton, R.C. Huang, Machine learning and clinical epigenetics: a review of challenges for diagnosis and classification. *Clin. Epigenetics* **12**, 1–11 (2020)
9. A. Jowaed, I. Schmitt, O. Kaut, U. Wllner, Methylation regulates alpha-synuclein expression and is decreased in Parkinson's disease patients' brains. *J. Neurosci.* **30**(18), 6355–6359 (2010)
10. U. Wllner, O. Kaut, L. deBoni, D. Piston, I. Schmitt, DNA methylation in Parkinson's disease. *J. Neurochem.* **139**, 108–120 (2016)
11. E. Miranda-Morales, K. Meier, A. Sandoval-Carrillo, J. Salas-Pacheco, P. Vzquez-Crdenas, O. Arias-Carrin, Implications of DNA methylation in Parkinson's disease. *Front. Mol. Neurosci.* **10**, 225 (2017)
12. E. Masliah, W. Dumaop, D. Galasko, P. Desplats, Distinctive patterns of DNA methylation associated with Parkinson disease: identification of concordant epigenetic changes in brain and peripheral blood leukocytes. *Epigenetics* **8**(10), 1030–1038 (2013)
13. Y.H. Chuang, K.C. Paul, J.M. Bronstein, Y. Bordelon, S. Horvath, B. Ritz, Parkinson's disease is associated with DNA methylation levels in human blood and saliva. *Genome Med.* **9**(1), 76 (2017)
14. A. Kakade, B. Kumari, P.S. Dholaniya, Feature selection using logistic regression in case-control DNA methylation data of Parkinson's disease: A comparative study. *J. Theor. Biol.* **457**, 14–18 (2018)

15. C. Wang, L. Chen, Y. Yang, M. Zhang, G. Wong, Identification of potential blood biomarkers for Parkinson's disease by gene expression and DNA methylation data integration analysis. *Clin. Epigenetics* **11**(1), 1–15 (2019)
16. Costanza L. Vallerga et al., Analysis of DNA methylation associates the cystine-glutamate antiporter SLC7A11 with risk of Parkinson's disease. *Nat. Commun.* **11**(1), 1–10 (2020)
17. T. Barrett, S.E. Wilhite, P. Ledoux, et al., NCBI GEO: archive for functional genomics data sets-update. *Nucl.C Acids Res.* **41**(D1), D991–D995 (2012)
18. T. Hastie, R. Tibshirani, G. Sherlock, M. Eisen, P. Brown, D. Botstein, Imputing missing data for gene expression arrays (1999)
19. X. Ma, Y.W. Wang, M.Q. Zhang, A.F. Gazdar, DNA methylation data analysis and its application to cancer research. *Epigenomics* **5**(3), 301–316 (2013)
20. P. Du, X. Zhang, C.C. Huang, N. Jafari, W.A. Kibbe, L. Hou, S.M. Lin, Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis. *BMC Bioinform.* **11**(1), 587 (2010)
21. M.E. Ritchie, B. Phipson, D.I. Wu, Y. Hu, C.W. Law, W. Shi, G.K. Smyth, limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucl. Acids Res.* **43**(7), e47–e47 (2015)
22. L. Yu, H. Liu, Feature selection for high-dimensional data: A fast correlation-based filter solution, in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)* (2003), pp. 856–863
23. I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines. *Mach. Learn.* **46**(1–3), 389–422 (2002)
24. D.H. Wolpert, Stacked generalization. *Neural Netw.* **5**(2), 241–259 (1992)
25. H.H. Hsu, C.W. Hsieh, M.D. Lu, Hybrid feature selection by combining filters and wrappers. *Expert. Syst. Appl.* **38**(7), 8144–8150 (2011)

Disease Detection and Prediction Using the Liver Function Test Data: A Review of Machine Learning Algorithms



Ifra Altaf, Muheet Ahmed Butt, and Majid Zaman

Abstract In the last decade, there has been an admirable improvement in the classification accuracy of various machine learning techniques used for disease diagnosis. This even aids in finding the associations and patterns in the data, which helps in the construction of prediction model. Diagnosing illness by considering the features that have the maximum impact on recognition is important to control the disease. The main objective of this research paper is to provide a summarized review of literature with comparative results, which has been done for the detection and prediction of liver diseases with various machine learning algorithms using the liver function test data in order to make the analytical conclusions. From this study, it is observed that the CMAC, RBF, PSO-LS-SVM and ADTree improve the accuracy of liver disease detection and prediction. A review of past findings on the LFT data and its association with diabetes prediction is also studied.

Keywords Liver function tests · Diabetes mellitus · Disease diagnosis · Deep learning · Artificial neural networks

1 Introduction

Large quantities of information about patients and their medical conditions are accumulated in the clinical databases. Data mining is used to extract knowledge by analyzing the disease datasets that consist of the symptoms of the particular disease under study. The extracted knowledge in terms of a useful pattern showing the correlation between various symptoms of the disease dataset acts as an information source for machine learning. Machine learning algorithms examine these patterns and learn

I. Altaf · M. A. Butt (✉)

Department of Computer Sciences, University of Kashmir, Srinagar, J&K, India

I. Altaf

e-mail: malikifra.scholar@kashmiruniversity.net

M. Zaman

DIRECTORATE OF IT&SS, UNIVERSITY OF KASHMIR, SRINAGAR, J&K, INDIA

from them so as to assist in predicting and diagnosing the disease. Data mining and machine learning promise an improved accuracy of perception and diagnosis of disease [1], thereby supporting the objectivity of decision-making process [2, 3]. The techniques have been widely and effectively used in other domains as well such as for educational data mining [4–7], geographical data mining [8–10], information translation [11], intrusion detection system [12, 13], etc.

Liver is the most complex body organ that plays an important role in many body functions. It occupies the upper right quadrant of the abdomen. Many essential tasks related to the metabolism are performed by the liver. The liver's ability to perform its metabolic, detoxification and storage functions is weakened as soon as the liver disease develops. In recent times, progressive studies have reported the relationship of chronic liver diseases with diabetes [14]. Abnormal liver function tests (LFTs) may also indicate other critical diseases that might benefit from earlier diagnosis, which may lead to more cures or longer survival [15]. The measurement of the enzymes in LFTs can indicate diabetes. Diabetes is a long-lasting chronic disease that influences the human body and is caused due to the expanded level of sugar fixation in the blood. There are three categories of diabetes, namely, type 1, type 2 and gestational diabetes [16]. Diabetes increases the risk of a condition where excess fat builds up in the liver—a condition known as nonalcoholic fatty liver disease (NAFLD) that occurs in at least half of those with type 2 diabetes. In the development of liver function impairment and in NAFLD, the insulin resistance is recognized as a pathological factor.

This paper is structured as follows: Sect. 2 briefs about the liver diseases, its diagnosis, liver function tests and the occurrence of deranged liver enzymes in diabetic patients. Section 3 introduces a comprehensive review of literature for liver disease diagnosis and prediction with machine learning techniques and systematic relationship between LFTs and diabetes disease in terms of related work. Section 4 gives the comparative study of summarized review and Sect. 5 discusses about the outcomes of comparative study.

2 Overview of Liver Diseases

Every year over 2.4% of Indian deaths occurs due to liver diseases [17]. In India, the burden of this disease is massive with 22.2 deaths/100,000 population attributed to one form of liver disease (cirrhosis) by the Global Health Observatory data from the World Health Organization [18]. More than a hundred different types of liver diseases have been identified, which can affect both males as well as females. The causes behind liver diseases are numerous stretching from infection, immune system abnormality, cancer and other growths to alcohol abuse and drug overdoses. The risk factors accompanying the liver disease are obesity, type 2 diabetes, exposure to body fluids and even the family history of having liver disease. Most commonly, the symptoms of liver disease include nausea, tiredness, anorexia, weakness, vomiting tendency, abdominal pain and abdominal swelling. Yellowish discolorations of the

Table 1 Tests and procedures for determining the liver diseases

Test	Description
Medical history and physical exam	The family history is queried and the clinical investigation such as palpation, auscultation, visual examination and percussion is done in order to understand the symptoms and the possible risk factors
Imaging tests	X-rays, magnetic fields or sound waves are used by the imaging tests to create pictures of the inside of the body. The imaging tests include ultrasound, computed tomography (CT) scan, magnetic resonance imaging (MRI), angiography to assist in finding the suspicious areas and spread of liver damage
Other tests and procedures	If the imaging outcomes are not sure, biopsy, i.e., the removal of a sample of tissue is done that includes needle biopsy, laparoscopic biopsy and surgical biopsy to ascertain the liver disease damage
Lab tests	To look for the signs of liver disease or to learn about the liver and general health, some biochemical tests such as a complete blood count test, liver function test or hepatitis serology blood test are done

eyes and skin, swollen gallbladder, weight loss, itching, pale stool, swollen spleen, swollen ankles, legs or abdomen are some other symptoms [19]. More than a hundred different types of liver diseases have been identified, which can affect both males as well as females out of which viral hepatitis, neonatal hepatitis, liver cancer, primary hepatoma, primary biliary cirrhosis, liver fibrosis, alcoholic liver damage, nonalcoholic liver disease, cholelithiasis, liver cirrhosis, primary sclerosing cholangitis, hemochromatosis, tyrosinemia and Wilson disease are usually prevalent [20].

2.1 Diagnosis of Liver Diseases

Liver diseases if left untreated can cause permanent damage but they can be manageable if detected earlier. The precise diagnosis of liver disease involves a clinical and diagnostic investigation performed by a health care professional. Liver disease can also be predicted by examining the enzymes—levels [21] in the blood. Table 1 shows the list of tests and ways for determining liver disease.

2.2 Liver Function Tests

Liver function tests (LFTs), also known as the hepatic panel, liver panel or liver function panel are among the most commonly performed blood tests that measure

Table 2 Different types of blood tests in the liver function panel

Test	Description
Bilirubin test	High levels in the blood could indicate liver damage or possibly jaundice
Alkaline phosphatase (ALP) test	High levels in the blood could indicate liver damage or disease, a blocked bile duct, or bone disease
Alanine transaminase (ALT) test	High levels in the blood can indicate liver damage or disease
Aspartate transaminase (AST) test	High levels in the blood can indicate liver damage or disease
Albumin and total proteins test	Low levels in the blood can indicate liver damage or disease
Gamma-glutamyl transferase (GGT) test	High levels in the blood can indicate liver damage or bile duct damage

the levels of proteins, liver enzymes and bilirubin in the blood, which help to determine the health of the human liver [22]. The damaged liver cells discharge enzymes into the bloodstream. A blood test can easily detect the deranged enzymes and hence the health of the liver. A liver function or hepatic function panel is done to acquire information about the levels of the total bilirubin, direct bilirubin, indirect bilirubin, aspartate aminotransferase also known as serum glutamic oxaloacetic transaminase (AST/SGOT), alanine aminotransferase also known as serum glutamic pyruvic transaminase (ALT/SGPT), alkaline phosphatase (ALP), gamma-glutamyl transpeptidase (GGT), albumin, globulin and the ratio of albumin and globulin (A:G Ratio) [20]. The predictive variables of the LFTs are the blood tests sensitive to liver disorders. There are many liver tests. Some common ones [23] are listed in Table 2. It may be an indication of liver disease if any of these tests are outside of the normal range.

2.3 Derangement of Liver Enzymes in Diabetes Mellitus

As stated by a growing body of research, diabetes shows a close correlation with hepatitis, liver cirrhosis and liver cancer. The co-occurrence of liver complications would additionally speed up the worsening of patients with diabetes. Liver cirrhosis and diabetes affect each other [24]. Liver is known as an insulin-sensitive organ. As liver regulates blood sugar, the excess fat can make it less responsive to insulin and hence too much glucose can remain in the blood that can cause diabetes mellitus [25]. India has become the diabetes capital of the world, surpassing the list of the 10 countries having the highest number of people with diabetes in 2000. India is suffering a frightening trend with 31.7 million diabetic people in the year 2000 to an estimation of 79.4 million diabetic people by 2030 [26].

3 Related Work

To date, enormous research work has been performed by various researchers for the diagnosis and prognosis of liver and diabetes diseases using different data mining as well as machine learning techniques.

3.1 Literature Survey of Liver Disease Detection and Prediction with Machine Learning Techniques

This section emphasizes some recent developments in liver function test data classification by assessing the dataset, algorithms, methods used by the authors and their observed results along with the future work to find out the efficient methods for medical diagnosis of diseases related to liver.

In 1989, Gabor [27] proposed a lazy model-based algorithm called DBPredictor that uses a greedy top-down search to locate a probabilistic IF-THEN rule for diagnosing liver disorders. The algorithm has the ability to avoid the need for the discretization of numerical attributes. The experimental results demonstrated empirically that the proposed method performs faster than the C4.5 algorithm as the impact of pruning has a constructive effect on the accuracy.

In 1994, Turney [28] introduced a new algorithm Inexpensive Classification with Expensive Tests (ICET) by hybridizing the genetic algorithm and a decision tree induction algorithm for cost-sensitive classification of alcoholic liver disorders and hepatitis. As per the results, the proposed algorithm is robust performed considerably better.

In 2002, Nong Ye et al. [29] presented a novel data mining algorithm, which they termed as clustering and classification algorithm-supervised (CCA-S). The algorithm is scalable as compared with the many existing data mining such as decision trees, artificial neural networks. The records with the missing values were removed from the dataset because the proposed model did not deal with the missing values. The model worked in two phases—training phase and testing phase. First, in the training phase, the data points of the training dataset were grouped into clusters. Next, the generated clusters were used to classify the testing dataset data points. As per the experimental results, the classification performance of CCA-S in terms of the error rate is comparable to the performance of base classifiers.

In 2003, Ozyilmaz et al. [30] tried diagnosing the hepatitis disease with the help of different neural networks. The neural network architectures used were standard feed-forward networks namely multilayer perceptron (MLP), radial basis function (RBF) and their hybrid network namely Conic Section function neural network (CSFNN). The authors trained the MLP with the standard back-propagation and trained the RBF with orthogonal least square algorithm. Also the CSFNN was trained with adaptive learning. For diagnosing the disease, fivefold cross-validation method has been used. The authors used the MATLAB neural network toolbox to implement

MLP and RBF, whereas the MATLAB code was written to implement CSFNN. The classification accuracy of MLP, RBF and CSFNN was recorded as 81.3%, 85% and 90%, respectively. Based on the experimental results, the proposed ensemble method showed the highest classification accuracy.

In 2004, Zhou et al. [31] explored and proposed a variant of C4.5 decision tree algorithm named NeC4.5 to diagnose liver disorder. In order to preprocess the training data for decision tree construction, NeC4.5 makes use of neural network ensemble. NeC4.5 shows a new way to hybrid learning and is different from earlier hybrid learning algorithms. A neural network ensemble is trained by the algorithm, which is then used to produce a new training set. The chosen class labels of the original training tuples are replaced with the output from the trained ensemble. The trained ensemble also produced some extra training tuples. These extra tuples were added to the new training set and then the C4.5 is trained using this new training set. The classification accuracy of the decision tree classifier is improved by the processed training data by neural network. The experiments reported in their paper showed that when an appropriate extra data ratio value is given, NeC4.5's generalization ability is better than that of C4.5.

In 2006, Revett et al. [32] classified and diagnosed the primary biliary cirrhosis using a probabilistic neural network (PNN)—an approach based on Bayes formula and Taylor's polynomial approximation. This algorithm used the concept of Rough Sets (RS) capacity to decrease the dimensionality of the dataset and also produce a set of easily comprehensible rules. The pre-processing stage of the dataset averages the values of the same test for the patients with multiple visits. The RS filtered and reduced the cardinality attributes and generated a very precise classifier when compared with the previous results. The authors proposed to use the idea of PNN because it can handle missing data items very well. Furthermore, it does not need the discretization of various types of data. The training accuracy for PNN by means of a separate smoothing factor for every classification outcome and for average number of dividing points as per the results was recorded as 99.9% while the testing accuracy is recorded as 83.80%.

In 2007, Comak et al. [33] developed a novel ensemble machine learning method that hybridized least square support vector machine (LSSVM) classifier with a new fuzzy logic-based weighting method called fuzzy weighting pre-processing to diagnose alcoholic liver disorder. As per the experimental results, the classification performance of the proposed method was attained as 94.29%. The area of ROC curves for standard LSSVM and for LSSVM with fuzzy weighting pre-processing is 0.336 and 0.95, respectively. The classification performance showed a considerable impact on the performance of classifiers by using the new fuzzy logic-based weighting method when compared with the existing methods.

In 2008, Neshat et al. [34] proposed a Fuzzy Expert System that included the fuzzy rules, triangular or trapezoidal fuzzifier and center of gravity defuzzifier formula to diagnose patients with healthy and unhealthy liver. The fuzzy system is utilized for analysis, learning and detection of liver disorders. The target variable deals with the rate of liver disorder risks. The performance accuracy for the proposed model based

on timely diagnosis of disease and assigning the rate of liver disorders considerably improved as per the previous methods and was recorded as 91%.

In 2009, Modjtaba et al. [35] used support vector machine (SVM), artificial neural networks and radial basis functions (RBFs) networks having two-layer structures with linear and Gaussian function that outperformed all the networks used previously in the literature. These techniques were employed for the diagnosis of Hepatitis diseases and to categorize the type and the phase of the disease. A performance comparison showed RBF as the best method selected for the diagnosis task with the overall accuracy was recorded as 96.4%.

In 2010, Bucak et al. [36] developed an expert diagnostic system and used the neural network approach called Cerebellar Model Articulation Controller (CMAC) for the diagnosis of healthy liver, hepatitis and cirrhosis. The liver data are first normalized into the range between 0 and 1. Then the CMAC artificial neural network is trained and tested with the normalized data through quantization, memory addressing. In order to find the weights, the difference between actual output and desired output was calculated. The classification accuracy for diagnosing the liver disease achieved with the design of CMAC ANN architecture was calculated as 100%. The authors were of the view that the use of other enzymes affecting the liver disease needs to be incorporated in order to diversify the calculated experimental results.

In 2011, Ming et al. [37] developed a fuzzy-based framework that utilized the k-means, unsupervised Gath-Geva and the Fast Global k-means supervised fuzzy clustering algorithms. The k-means algorithm was used to determine the actual number of cluster required for different data sets and the fast global k-means algorithm was needed to improve the computation time taken by global k-means algorithm. The supervised fuzzy clustering algorithms handle the small size noisy data effectively. The authors have implemented these algorithms on the hepatobiliary disorder dataset. The experimental results show that the classification rate for classifying the hepatobiliary disorder with enhanced supervised fuzzy clustering using the tenfold validation as 58.78%.

In 2012, Bendi et al. [38] tried to use the modified rotation forest (MRF) algorithm made by the combination of multilayer perception classification algorithm with random subset for classifying the liver disease patients. The authors used the principal component analysis (PCA); correlation-based feature selection (CFS), random projection and random subset feature selection methods to get the best combination of the algorithms. The experimental results show that 73.07% accuracy is given by the modified rotation forest on the LFT dataset using the tenfold cross-validation.

In 2013, Novita et al. [39] put forward the ensemble of decision tree and Naïve Bayes known as Naïve Bayes decision tree (NB Tree) for optimizing the liver disease classification. The study compared the decision tree, Naïve Bayes and NB tree algorithms on the liver function test dataset. The experimental results showed that the proposed hybrid algorithm recorded the highest accuracy score of 67.01% among the three classifiers. Conversely, the Naïve Bayes algorithm recorded the fastest execution time.

In 2014, Soliman et al. [40] put forward a hybrid algorithm to classify the hepatitis C virus. The study suggested an ensemble classification system by combining the principal component analysis algorithm (PCA), modified particle swarm optimization algorithm (MPSO) and least squares support vector machine algorithm (LS-SVM) to form the novice PCA-PSO-LS-SVM classifier to diagnose the hepatitis C virus among patients using the hepatitis dataset. The proposed system consisted of four phases where the first phase was to preprocess the data using the local mean method and then the PCA algorithm was employed which extracted the most influential attributes from the dataset. The attributes were reduced to six. It was followed by the optimization of parameters using the PSO algorithm and then the optimized parameters were fed to the LS-SVM classifier to predict whether the patient with virus lives or dies. Using the tenfold cross-validation method, the classification accuracy of LS-SVM came out to be 96.12% whereas that of the proposed system was 98.86% and thus it revealed the superiority of the proposed system.

In 2015, Ayeldeen et al. [41] tried to predict the liver fibrosis stages by using the hepatitis C dataset of Egyptian patients that contained the values of serum biomarkers with the help of a decision tree classifier. The target variable was categorized into four classes ranging from absence of fibrosis to presence of fibrosis. Using the appropriate feature selection and correlation equation with the decision tree classifier, the classification accuracy was calculated as 93.7%.

In 2016, Birjandi et al. [42] paper attempted to predict and diagnose the non-alcoholic fatty liver disease by using a non-parametric statistical learning approach. The authors used the classification tree approach to identify the factors associated with the disease in the area of Iran. From the experimental results, the prediction accuracy to predict fatty liver occurrence was 80% for training data and 75% for testing data. Additional noteworthy point of the study was the finding of an association between metabolic syndrome and non-alcoholic fatty liver disease.

In 2017, Mafazalyaqeen et al. [43] predicted the liver disease using the Boosted C5.0 classification and Genetic Algorithm (BC5.0-GA). For the production of rules, the authors used the GA instead of the evolutionary algorithm. The proposed model optimized the rules and the proposed model was analyzed in MATLAB so as to implement the genetic algorithm on the rules. The experimental results show the accuracy of Boosted C5.0 separately as 81.87% and that of the proposed ensemble method as 92.93%, which revealed that the optimized model can better predict or diagnose the disease.

In 2018, Mohaimenul et al. [44] tried to select the prognostic factors with the help of random forest (RF), support vector machine (SVM), artificial neural network (ANN) and logistic regression (LR) data mining techniques to predict the fatty liver disease. The proposed model consisted of seven steps that included a feature selection approach and tenfold cross-validation to recurrently screen potential variables. The dataset was preprocessed using imputation and normalization to get a high-quality dataset. The model was developed and validated in the Weka. As per the experimental results, the logistic regression technique provides a better result in the prediction of fatty liver diseases with an accuracy of 76.30%.

In 2019, Masaya et al. [45] attempted to put forward gradient boosting (GB) novel predictive model for the diagnosis of hepatocellular carcinoma. The diagnostic blood test values of liver function test (LFT) were used for the study. Using the optimal hyperparameter, the predictive accuracy of gradient boosting for determining the presence of hepatocellular carcinoma was calculated as 87.34%.

In 2020, Somaya et al. [46] developed Alternating Decision Tree (ADTree) prediction model, one class of decision tree learning that combined boosting and decision tree algorithms for diagnosing the chronic hepatitis C virus-related hepatocellular carcinoma with predictive accuracy 95.6%. Experimental results showed that ALP, albumin and total bilirubin of the LFT were statistically associated with hepatocellular carcinoma presence.

3.2 Literature Survey for Prevalence of Abnormal Liver Function Tests in Diabetes Mellitus

According to the previous research work, diabetes has an effect on the liver and there is a valid association between the liver function enzymes and onset of diabetes. The liver enzymes are usually deranged in diabetic people and act as the indicators of hepatocellular injury [47]. Some of the recent research papers that show this relationship are as follows:

In 2011, Nguyen et al. [48] put forward that the elevation in the markers of liver dysfunction and nonalcoholic fatty liver are believed to be a part of metabolic syndrome and are related to diabetes. The study alleged that ALT/SGPT and GGT are the potential biomarkers for diabetes risk assessment. The values for area under the receiver operating curve regarding the predictive value of ALT and GGT were significantly higher for diabetes as compared to pre-diabetes.

In 2012, Han Ni et al. [49] tried to establish the relationship between the liver function test abnormalities and blood glucose in diabetic patients in Myanmar. The authors comprehended that the ALT, AST and bilirubin were raised by 18.5%, 14.8% and 4.9%, respectively, in diabetic people. There was no significant correlation of age, family history of diabetes, mode of therapy or type of diabetes with the mean values of ALT and AST. The elevated and deranged ALT and AST are the markers for related non-alcoholic fatty liver disease in diabetes patients.

In 2013, Hadi Salih et al. [50] performed the statistical analysis of the liver function tests using the excel software and showed that the individuals having type 2 diabetes had a higher incidence of liver function test abnormalities. The research study showed that the glucose levels were positive significantly correlated with ALP, ALT and AST in females.

In 2014, Philip et al. [47] alleged that the liver enzymes can be utilized as biomarkers for the assessment of diabetes. The authors claimed that there is a considerable increased activity of AST, ALT and GGT in diabetic dataset which is related to insulin resistance and hence type 2 diabetes. The relationship between the serum

glucose and liver function enzymes was evaluated on the basis of the Spearman's correlation calculations. The results of the study showed that the ALT, AST and GGT levels in type 2 diabetes patients increased considerably.

In 2016, Kaustubh et al. [51] tried to specify the incidence of concurrent derangements of liver enzymes in diabetic population from Shillong, Meghalaya. The authors tried to find the association between the fasting serum glucose and the following tests in the LFT panel namely total bilirubin, AST, ALT, ALP and albumin using the multivariate linear regression analysis. The results showed that 71.25% of the patient records had deranged test values in at least one LFT. The ALT and ALP were elevated by 46.8% and 48.5% respectively. In both male and female patient records, the serum ALP correlated positively with fasting glucose.

In 2018, Ghimire et al. [52] tried to study the abnormalities in the liver function parameters and glycemic status among diabetic Nepalese population. To diagnose the diabetes mellitus in the population, fasting blood sugar (FBS) and HbA1c were assessed. Using the SPSS version 20.0, the parameters of LFT were analyzed and it was observed that there is an increased level of ALT by 57%, AST by 46% and ALP among patients with diabetes mellitus. Though, total protein, albumin and A/G ratio were considerably decreased in diabetic group of patients. There was a positive correlation between levels of HbA1c and chronic changes in transaminases at a substantial level.

In 2019, Aditi et al. [53] found that the AST/SGOT, ALT/SGPT, ALP and bilirubin were elevated with the frequencies of 59.3%, 52.6%, 42.1% and 31.5% respectively in type 2 diabetic dataset of north Indian population. The frequency of diminished albumin was 73.6% in type 2 diabetes. The study further showed that the ALP concentration correlated with the blood sugar fasting levels significantly.

Again in 2019, Getnet Teshome et al. [54] used the systematic random sampling technique, binary logistic regression and bivariate correlation on the liver function tests (LFTs), lipid profiles and fasting blood sugar to check the prevalence of abnormal LFTs in type 2 diabetes mellitus. The authors found out that the ALT was the most frequently raised liver enzyme in diabetic people and 33.3% of the diabetic dataset had one or more liver test abnormality.

In 2020, Alampally et al. [55] conducted a study to show the incidence of liver function impairment with diabetes. The study showed that there is an association of serum liver markers such as AST, ALT, ALP, total bilirubin and total proteins with random blood sugar in diabetic people. The elevated ALT was found to be the most common abnormality.

4 Comparative Analysis of Various Machine Learning Algorithms Using LFT for Diagnosing and Predicting Liver Diseases

Based on the accuracies, the comparative performance of various machine learning techniques for the detection and prediction of chronic liver disease using the predictive variables from the liver function test have been summarized in Table 3.

Figure 1 gives the graphical representation about the accuracies of the various machine learning algorithms that have been reviewed in this paper and intended for liver disease diagnosis and prognosis by using the values of the constituent blood tests of LFT sensitive to liver disorders.

5 Discussion

The machine learning techniques have overpowered the statistical models as the former produce good performance results and successfully deal with the categorical data, missing values and large data points. One of the major concerns of researchers for developing the detecting and predicting models is the accuracy. The accuracy is directly proportional to the choice of research tools and techniques. The study shows a review of recent research on liver disease detection and prediction using various data mining and machine learning techniques and algorithms. The research shows that different experimental tools such as Weka, MATLAB, Scikit-Learn, etc., are used to predict diseases. Also, different liver disorder datasets have been used in different experiments that consisted of either the entire or constituent test values of the liver function panel along with some other clinical, demographic and diagnostic biochemical attributes. In this study, the survey shows that the ensemble algorithms such as CMAC, RBF, PSO-LS-SVM and ADTree are suitable for predicting and detecting liver diseases as compared with the base classifiers like PNN and logistic regression. A systematic effort of identifying, selecting and evaluating the past work inferred that artificial neural networks have a wide acceptance and have obtained higher accuracy results. There is no such exclusive algorithm to predict or detect all kind of diseases. The study shows that the combination of different techniques to form the hybrid models or ensembles can be very promising in the efficient diagnosis of various liver diseases. This study also found that the type of dataset and pre-processing techniques applied to it is also important for good prediction results. The quality of the data with impactful and minimum predictive attributes by employing proper feature selection methods can assist a clinician with the effective diagnosis, preventive and therapeutic resolution to ease the worldwide burden of liver diseases.

This review emphasizes the association of diabetes mellitus with liver function abnormalities. The study shows that the diabetes disease can induce the liver function impairment by elevating mostly the ALT, AST and GGT levels in blood and hence justifies a concrete pathological relationship between liver and diabetes.

Table 3 Comparative analysis of various machine learning algorithms using LFT

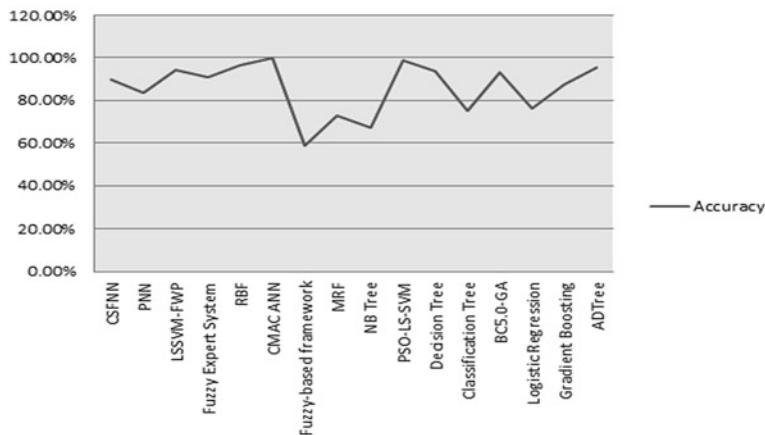
Author, year	Attributes used from LFT	Disease dealt	Techniques used	Accuracy (%)
Melli, 1989	ALP, ALT(SGPT), AST(SGOT) and GGT	General liver disorders	DBPredictor	–
Turney, 1994	ALP, ALT, AST, GGT and albumin	Alcoholic liver disorder, hepatitis	ICET	–
Nong Ye et al., 2002	ALP, ALT, AST and GGT	General liver disorders	CCA-S	–
Ozyilmaz et al., 2003	Bilirubin, ALP, AST and albumin	Hepatitis	CSFNN	90
Zhou et al., 2004	ALP, ALT, AST, and GGT	General liver disorders	NeC4.5	–
Revett et al., 2006	Bilirubin, albumin, ALP and AST	Primary biliary cirrhosis	PNN	83.80
Comak et al., 2007	ALP, ALT, AST and GGT	Alcoholic liver disorder	LSSVM	94.29
Neshat et al., 2008	ALP, ALT, AST, GGT	General liver disorders	Fuzzy expert system	91
Modjtaba et al., 2009	AST, ALT, ALP, GGT, total bilirubin, direct bilirubin and albumin	Hepatitis	RBF	96.4
Bucak et al., 2010	AST, ALT, AST/ALT, albumin and protein	Hepatitis and cirrhosis	CMAC ANN	100
Ming et al., 2011	AST, ALT, ALP, bilirubin	Alcoholic liver damage, primary hepatoma, liver cirrhosis, cholelithiasis	Fuzzy-based framework	58.78
Bendi et al., 2012	ALP, ALT, AST and GGT	General liver disorders	MRF	73.07
Novita et al., 2013	AST, ALT, ALP, GGT, total bilirubin, direct bilirubin, globulin, albumin and A:G ratio	General liver disorders	NB Tree	67.01
Soliman et al., 2014	AST, ALP, bilirubin, albumin	Hepatitis C	PSO-LS-SVM	98.86
Ayeldeen et al., 2015	AST, ALT, GGT, total bilirubin, direct bilirubin, albumin and globulin	Liver fibrosis stages in hepatitis C	Decision tree	93.7

(continued)

Table 3 (continued)

Author, year	Attributes used from LFT	Disease dealt	Techniques used	Accuracy (%)
Birjandi et al., 2016	ALT, AST and albumin	Non-alcoholic fatty liver disease	Classification tree	75
Mafazalyaqeen et al., 2017	AST, ALT, ALP, GGT, total bilirubin, direct bilirubin, albumin, globulin and A:G ratio	General liver disorders	BC5.0-GA	92.93
Mohaimenul et al., 2018	ALP, AST, ALT, GGT, total bilirubin, direct bilirubin, albumin and globulin	Fatty liver disease	Logistic regression	76.30
Masaya et al., 2019	AST, ALT, ALP, GGT, total bilirubin and albumin	Hepatocellular carcinoma	Gradient boosting	87.34
Somaya et al., 2020	AST, ALT, ALP, total bilirubin and albumin	Chronic hepatitis C virus related hepatocellular carcinoma	ADTree	95.6

Performance of Various Classifiers

**Fig. 1** Comparative graph of classifiers for diagnosing/predicting liver diseases

6 Conclusion

There is a key research space for the classification of medical dataset for correct diagnosis. The hybrid models can be enhanced for more accurate prediction of liver diseases as well as for decision-making. There is a need for implementing more deep neural network algorithms on the liver function test dataset and compare the results with the classical approaches to look for the likelihood of improving the accuracy of disease prediction. The comparative study focuses on the fact that there is a need for a method that can automatically update the trained models whenever the new data gets available.

The researchers consider that liver problems lead to diabetes and contrariwise. Therefore, the deranged liver enzymes or deranged LFTs can be used to predict diabetes disease. Further researches in terms of the predictive effectiveness of the liver enzymes for diabetes with the help of machine learning techniques are needed to validate the findings. The review highpoints the significance of liver function monitoring in patients with diabetes and can be helpful for motivating clinicians to observe the neglected hepatic dysfunction in diabetes mellitus. This can assist physicians to effectively identify the preventive and therapeutic ways to moderate the global burden of liver and diabetes diseases.

References

1. P. Sharma, et al., Diagnosis of Parkinson's disease using modified grey wolf optimization. *Cogn. Syst. Res.* **54**, 100–115 (2019)
2. M. Ashraf, et al., Prediction of cardiovascular disease through cutting-edge deep learning technologies: an empirical study based on TENSORFLOW, PYTORCH and KERAS, in *International Conference on Innovative Computing and Communications* (Springer, Singapore, 2020)
3. J.A. Alzubi, et al., Efficient approaches for prediction of brain tumor using machine learning techniques. *Indian J. Public Health Res. Dev.* **10**(2), 267–272 (2019)
4. M. Ashraf, M. Zaman, M. Ahmed, An intelligent prediction system for educational data mining based on ensemble and filtering approaches. *Procedia Comput. Sci.* **167**, 1471–1483 (2020)
5. M. Ashraf, Z. Majid, A. Muheet, To ameliorate classification accuracy using ensemble vote approach and base classifiers, in *Emerging Technologies in Data Mining and Information Security* (Springer, Singapore, 2019), pp. 321–334
6. M. Ashraf, Z. Majid, A. Muheet, Performance analysis and different subject combinations: An empirical and analytical discourse of educational data mining, in *2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (IEEE, 2018)
7. M. Ashraf, M. Zaman, M. Ahmed, Using Ensemble StackingC method and base classifiers to ameliorate prediction accuracy of pedagogical data. *Procedia Comput. Sci.* **132**, 1021–1040 (2018)
8. R. Mohd, A.B. Muheet, Z.B. Majid, GWLM–NARX. *Data Technol. Appl.* (2020)
9. R. Mohd, A.B. Muheet, Z.B. Majid Baba.SALM-NARX: Self Adaptive LM-based NARX model for the prediction of rainfall, in *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2018 2nd International Conference on. IEEE, 2018.

10. Z. Majid, K. Sameer, A. Muheet, Analytical comparison between the information gain and Gini index using historical geographical data|| (IJACSA) Int. J. Adv. Comput. Sci. Appl. **11**(5), 429–440 (2020)
11. M. Zaman, S.M.K. Quadri, A.B. Muheet, information translation: a practitioners approach. Proc. World Congr. Eng. Comput. Sci. **1** (2012)
12. A. Omar, Deep learning-based intrusion detection model for industrial wireless sensor networks. J. Intell. Fuzzy Syst. (2020), In press
13. N.M. Mir, et al., An experimental evaluation of bayesian classifiers applied to intrusion detection. Indian J. Sci. Technol. **9**(12), 1–7 (2016)
14. Y. Zhao, X. Huichun, A different perspective for management of diabetes mellitus: controlling viral liver diseases. J. Diabetes Res. **2017** (2017)
15. D.J. McLernon, et al., The utility of liver function tests for mortality prediction within one year in primary care using the algorithm for liver function investigations (ALFI). PLoS One **7**(12), e50965 (2012)
16. C. Kalaiselvi, G.M. Nasira, A new approach for diagnosis of diabetes and prediction of cancer using ANFIS, In *2014 World Congress on Computing and Communication Technologies* (IEEE, 2014)
17. S. Sontakke, L. Jay, D. Reskul, Diagnosis of liver diseases using machine learning, in *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)* (IEEE, 2017)
18. M. Jain, et al., Incidence and risk factors for mortality in patients with cirrhosis awaiting liver transplantation. Indian J. Transplant. **13**(3), 210 (2019)
19. A. Ifra, A.B. Muheet, Z. Majid, S. Jahangir Sidiq, A comparative study of various data mining algorithms for effective liver disease diagnosis a decade review from 2010 to 2019. **6**(1), 980–995 (2019)
20. Diseases and Conditions, Apollo Hospitals, <https://www.apollohospitals.com/patient-care/heal-th-and-lifestyle/diseases-and-conditions>
21. A. Koch, Schiff's diseases of the liver—10th edition. J. Am. Coll. Surg. (2007)
22. MedlinePlus, U.S. National Library of Medicine, <https://medlineplus.gov/lab-tests/liver-function-tests/>
23. Liver Function Test, <https://www.webmd.com/hepatitis/liver-function-test-lft>
24. Y. Zhao, et al., Management of diabetes mellitus in patients with chronic liver diseases. J. Diabetes Res. **2019** (2019)
25. The Hidden Risk of Liver Disease From Diabetes, WebMD, <https://www.webmd.com/diabetes/diabetes-liver-disease-hidden-risk>
26. S. Wild, et al., Global prevalence of diabetes: estimates for the year 2000 and projections for 2030. Diabetes Care **27**(5), 1047–1053 (2004)
27. G. Melli, *A Lazy Model-Based Approach to On-Line Classification* (Simon Fraser University, 1998)
28. P.D. Turney, Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. J. Artif. Intell. Res. **2**, 369–409 (1994)
29. N. Ye, X. Li, A scalable, incremental learning algorithm for classification problems. Comput. Ind. Eng. **43**(4), 677–692 (2002)
30. L. Ozyilmaz, Y. Tulay, Artificial neural networks for diagnosis of hepatitis disease, in *Proceedings of the International Joint Conference on Neural Networks*, vol. 1 (IEEE, 2003)
31. Z.-H. Zhou, Y. Jiang, NeC4. 5: neural ensemble based C4. 5. IEEE Trans. Knowl. Data Eng. **16**(6), 770–773 (2004)
32. K. Revett, et al., Mining a primary biliary cirrhosis dataset using rough sets and a probabilistic neural network, in *2006 3rd International IEEE Conference Intelligent Systems* (IEEE, 2006)
33. E. Comak, et al., A new medical decision making system: least square support vector machine (LSSVM) with fuzzy weighting pre-processing. Expert. Syst. Appl. **32**(2), 409–414 (2007)
34. M. Neshat, et al., Fuzzy expert system design for diagnosis of liver disorders, in *2008 International Symposium on Knowledge Acquisition and Modeling* (IEEE, 2008)
35. M. Rouhani, M. Motavalli Haghghi, The diagnosis of hepatitis diseases by support vector machines and artificial neural networks, in *2009 International Association of Computer Science and Information Technology-Spring Conference* (IEEE, 2009)

36. İÖ Bucak, S. Baki, Diagnosis of liver disease by using CMAC neural network approach, *Expert. Syst. Appl.* **37**(9), 6157–6164 (2010)
37. L.M. Ming, L. Chu Kiong, L.W. Soong, Autonomous and deterministic supervised fuzzy clustering with data imputation capabilities. *Appl. Soft Comput.* **11**(1), 1117–1125 (2011)
38. B.V. Ramana, M.S. Prasad Babu, N.B. Venkateswarlu, Liver classification using modified rotation forest. *Int. J. Eng. Res. Dev.* **6**(1), 17–24 (2012)
39. S.N.N. Alrifisahrin, T. Mantoro, Data mining techniques for optimization of liver disease classification, in *2013 International Conference on Advanced Computer Science Applications and Technologies* (IEEE, 2013)
40. O.S. Soliman, E.A. Elhamd, Classification of hepatitis C virus using modified particle swarm optimization and least squares support vector machine. *Int. J. Sci. Eng. Res.* **5**(3), 122 (2014)
41. H. Ayeldeen, et al., Prediction of liver fibrosis stages by machine learning model: A decision tree approach, in *2015 Third World Conference on Complex Systems (WCCS)* (IEEE, 2015)
42. M. Birjandi, et al., Prediction and diagnosis of non-alcoholic fatty liver disease (NAFLD) and identification of its associated factors using the classification tree method. *Iran. Red Crescent Med. J.* **18**(11) (2016)
43. M. Hassoon, et al., Rule optimization of boosted c5. 0 classification using genetic algorithm for liver disease prediction, in *2017 International Conference on Computer and Applications (ICCA)* (IEEE, 2017)
44. M.M. Islam, et al., Applications of machine learning in fatty live disease prediction. MIE (2018)
45. M. Sato, et al., Machine-learning approach for the development of a novel predictive model for the diagnosis of hepatocellular carcinoma. *Sci. Rep.* **9**(1), 1–7 (2019)
46. Hashem, Somaya, et al. “Machine Learning Prediction Models for Diagnosing Hepatocellular Carcinoma with HCV-related Chronic Liver Disease.” *Computer Methods and Programs in Biomedicine* (2020): 105551.
47. R. Philip, M. Mathias, K.M. Damodara Gowda, Evalation of relationship between markers of liver function and the onset of type 2 diabetes. *J. Health Allied Sci.* **4**(2), 090-093 (2014)
48. Q.M. Nguyen, et al., Elevated liver function enzymes are related to the development of prediabetes and type 2 diabetes in younger adults: the Bogalusa Heart Study. *Diabetes Care* **34**(12), 2603–2607 (2011)
49. H. Ni, H.H.K. Soe, A. Htet, Determinants of abnormal liver function tests in diabetes patients in Myanmar. *Int J Diabetes Res* **1**(3), 36–41 (2012)
50. D.H. Salih, Study of liver function tests and renal function Tests in diabetic type II patients. *IOSR J. Appl. Chem* **3**(3), 42–44 (2013)
51. K. Bora, et al., Presence of concurrent derangements of liver function tests in type 2 diabetes and their relationship with glycemic status: a retrospective observational study from Meghalaya. *J. Lab. Physicians* **8**(1), 30 (2016)
52. S. Ghimire, et al., Abnormal liver parameters among individuals with type 2 diabetes mellitus Nepalese population. *Biochem Pharmacol (Los Angel)* **7**(1), 2167-0501 (2018)
53. A. Singh, et al., Deranged liver function tests in type 2 diabetes: a retrospective study
54. G. Teshome, et al., Prevalence of liver function test abnormality and associated factors in type 2 diabetes mellitus: a comparative cross-sectional study. *EJIFCC* **30**(3), 303 (2019)
55. D. Nikitha Alampally, DS Jaipuriar, N. Alampally, A study on liver function impairment in type-2 diabetes mellitus. *IJRAR-Int. J. Res. Anal. Rev. (IJRAR)* **7**(1), 939–943 (2020)

Performance Assessment of Health Decision-Making Using Various Artificial Intelligence Techniques and Evolutionary Algorithms



Prabhav Jain, Ekansh Chauhan, and Varun Goel

Abstract Presently, the latest technology is used for health management and diagnostic strategy in the well-being area. Artificial intelligence usually helps in medical problems utilizing various models, machine learning typically helps to make decisions about health problems. A plethora of help is provided in the prediction of diseases. Different machine learning techniques are available to classify and identify diseases but what comes first is the optimization of the techniques. Nature-based optimization techniques are developed, which works on the way things interact with each other in nature. In this paper, particle swarm optimization (PSO) is used to optimize the dataset on diseases such as cardiovascular disease, liver disease and cancer. Along with PSO, principal component analysis (PCA) is used to optimize as well as decrease the computational complexity of the model. The key aim is to find the most important attributes contributing to disease such that early diagnosis and hence treatment of the disease can be started. The proposed hybrid model is trained using the data sets for different diseases with different classification algorithms viz random forest, support vector machine (SVM) and K-nearest neighbour (KNN). Experimental results show that all the irrelevant features that were not necessary were removed by particle swarm optimization and principal component analysis. The highest classification accuracy of 83.52% was obtained for Heart Disease dataset with random forest classifier. The highest classification accuracy of 81.32% was obtained for Liver Disease dataset with Random Forest classifier. Highest classification accuracy of 100% was obtained for Cancer dataset with random forest classifier as well as support vector machine classifier.

Keywords Particle Swarm optimization · Machine learning · Health · Cancer · Disease prediction · Nature-inspired optimization

P. Jain (✉) · E. Chauhan · V. Goel
Maharaja Agrasen Institute of Technology, Delhi, India

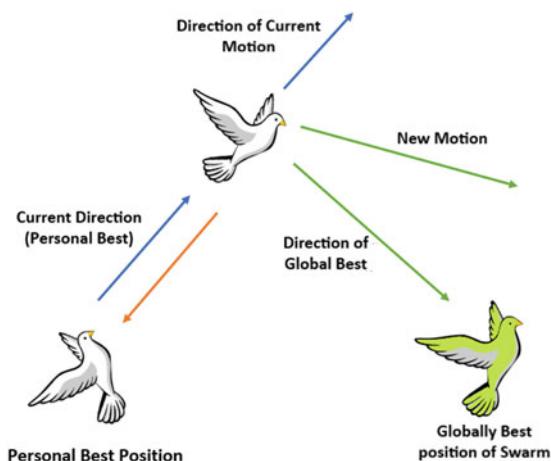
V. Goel
e-mail: varungoel@mait.ac.in

1 Introduction

There are various types of diseases and disorders due to which there are a lot of deaths. Some of the most common diseases include cardiovascular disease, cancer and liver disease. These diseases are the leading cause of death in various countries. Studies show that one in four people die due to cardiovascular disease and mainly due to heart disease or any heart disorder. Approximately, 2 million deaths per year worldwide are due to liver diseases. The second deadliest disease in the world, cancer causes an estimated 9.6 million deaths, which is equivalent to 1–deaths. Various tests are conducted in order to predict if a person is suffering from any of these diseases. In case of critical condition of a person, it becomes important to identify the right problems and factors affecting the person. The major problems should be prioritized and the minor ones should be taken care of, so that immediate medication can be provided to the patient. Considering an immediate solution to this problem can be predicted through machine learning algorithms. Machine learning can be seen as an emerging field for prediction if the amount of data is sufficient in order to train a model [1]. It can make conclusions from data easily which are impossible or extremely difficult for humans. The objective of the paper is to find efficient techniques and to find the most important parameters governing the deterioration of the health of a person. The key findings of the paper are to compare different machine learning algorithms and to compare the accuracy of all these by using a nature-based optimization algorithm. Manual data were taken in this article from various sources consisting of different diseases. The dataset was used in a way such that 80% was trained and 20% was used for testing.

Nature-based algorithms are made such that they use the natural phenomenon of the movement or any unique characteristic occurring in nature. This helps in uniquely identifying which features are more desirable and which are less. In this article, particle swarm optimization (PSO) [2] is used for optimization. Figure 1 shows

Fig. 1 Particle swarm optimization technique



movement of swarms in PSO in order to select optimal attributes. This particular algorithm is chosen because of its applicability on various datasets as well as its simplicity of usage. Along with PSO, principal component analysis (PCA) [3] is used. PCA is an unsupervised learning technique in machine learning. It is also used in feature reduction. Also, it is used to reduce the computational complexity, so that the speed of machine learning algorithms increases. PCA is mainly used with PSO because generally single optimization technique leads to low accuracy and becomes difficult to generalize some complicated problems. After the optimization of the dataset, various machine learning algorithms are used viz random forest, support vector machine and K-nearest neighbour. According to the previous research studies, it was found that various optimization techniques were used but the focus was not on the accuracy of machine learning model. In this research, the mere focus was to improve the accuracy of the model. Many researchers have also tried to use hybrid optimization algorithms [5]. In addition, researchers have combined the optimization with standard machine learning algorithms such as random forest, support vector machine, Naïve Bayes or logistic regression.

2 Related Work

According to [4], an artificial neural network (ANN) model is developed to predict whether a patient has any liver disease or not. Indian Liver Patient disease dataset was used for this purpose. JustNN tool was used to develop the model. On a dataset comprising 583 liver patients, the model was trained. The important factors affecting the person's health were identified and a validation accuracy of 99% was obtained.

According to [5], a hybrid model construction is proposed for improving accuracy of prediction of the presence of liver disease in patients. The research is conducted in three phases. First, data are collected from UCI Machine Learning Repository and classification algorithms are applied to it. In the second phase, a subset of data is taken containing only relevant features and then classification algorithms are applied on it. Last, all the results are compared with each other. The best result was found by using random forest classification with feature selection and an accuracy of 71.8696% is obtained.

According to [6], a prediction system for breast cancer at an early stage is developed. The research has been carried out by using the Wisconsin breast cancer dataset (WBCD). The smallest set of attributes is selected and then analyzed. A maximum of 99.28% of classification accuracy is achieved by this model.

According to [7], a knowledge-based system is proposed for breast cancer classification. For clustering, expected maximization is used. This is used to cluster the data in similar groups. For the classification, the fuzzy rules are generated using classification and regression trees (CART). Principal component analysis (PCA) is used to reduce computational cost of the model. The proposed knowledge-based system increases the accuracy of the prediction of breast cancer to a large extent.

According to [8], a model for prediction of heart disease is developed using machine learning algorithms such as Naïve Bayes, support vector machine, K-nearest neighbour. To predict Heart Disease using UCI dataset for heart disease prediction, several experiments were conducted. An accuracy of 82.17% and 84.28% is obtained using the Naïve Bayes classification technique for both cross-validation and train-test split techniques, respectively.

According to [9], a hybrid approach is proposed for the prediction of heart disease. A framework is set up by combining different data mining classifiers with novel classifiers. For evaluation and prediction, UCI heart disease dataset is used. An accuracy of 82% was obtained, which was found to be higher as compared with the individual techniques.

3 Methodology

In this research, along with particle swarm optimization (PSO), principal component analysis (PCA) is used. With the functioning of both the optimization techniques, the accuracy of the model is increased as well as the computational complexity is also decreased. Second, a major focus was to choose the right parameters for the different machine learning algorithms so that there is least deviation from the accurate results. Hyperparameter tuning is used for each of the machine learning algorithm. The parameters which gave the best accuracy are chosen and the model is trained accordingly. Figure 2 shows the step-by-step flow chart of the research being done on different datasets.

The main objective of this research is the prediction of three diseases namely heart disease, cancer and diabetes using particle swarm optimization and principal component analysis.

3.1 Datasets

The data were collected from UCI Machine Learning Repository as well as Kaggle for three diseases viz heart disease [10], cancer [11] and liver disease [12]. The dataset acquired was split in a way such that 80% was used for training and 20% for testing.

The first disease is heart disease and its dataset is named as Heart disease dataset, which can be found in the UCI Machine Learning Repository [13]. This dataset consists of 14 different attributes and consists of a few missing values. This dataset was created by four institutes. In this research, the Cleveland database is used. This dataset is widely used by researchers for any research related to heart. For fast and efficient processing of data, the ‘Sex’ attribute was removed. The dataset contained certain missing values. So, a mean was calculated for each attribute and was substituted in the missing values if it contained any. This was the first step of preprocessing.

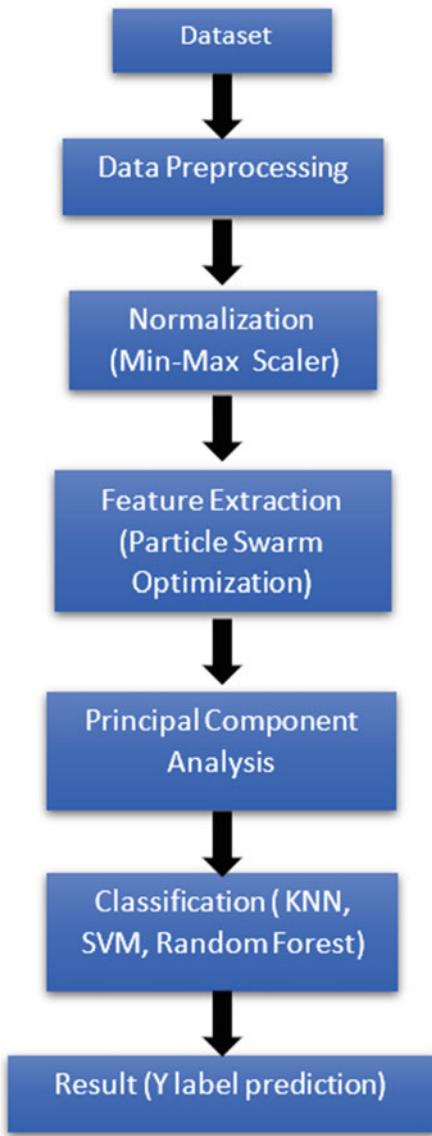


Fig. 2 Process flow chart

After this, the whole dataset was normalized so that it was in a consistent range between 0 and 1.

The second disease is liver disease and its dataset is named as Indian Liver Patient Dataset, which can be found in the UCI Machine Learning Repository. This dataset consists of 583 instances for 10 different attributes. This dataset is collected from

Indian subcontinent particularly from Andhra Pradesh. This particular dataset is chosen because it consists of zero missing values with the help of which accurate results can be predicted. In this dataset, the gender attribute was removed and then the dataset was normalized.

The third disease is cancer and can be found on Kaggle. It consists of 1000 records of patients spread over 23 attributes. The dataset was categorized on the intensity of the disease spread. It was either low, medium or high represented by 0, 1 and 2. The dataset consisted of no missing values and directly normalization was applied to keep the values within a specified range.

The research was carried in order to eliminate the least important factors and to predict accurate results. All the preprocessing was done one after another on the three datasets and was then taken for further processing.

3.2 *Feature Selection*

3.2.1 Particle Swarm Optimization

Feature selection is basically an optimization technique, which is used to incorporate the best available features and to remove all those redundant and irrelevant features, which may affect any training ability of the model [2]. There are an enormous number of problems in science and technology, which require optimization techniques. The solution to the nonlinear problems is difficult to optimize using traditional algorithms. A recent innovation to this problem is to use nature-based algorithms. These algorithms work on the principle of nature specifically animals. These algorithms eliminate all the irrelevant features and thereby saving the computation cost and also helps in determining accurate results. One of the nature-based optimization techniques is particle swarm optimization. Particle swarm optimization was found in 1995 by Russel Eberhart, an engineer and James Kennedy, socio-psychologist who were inspired by the living world. The nature-based algorithms generally use a swarm of multiple agents, which interact with each other to generate search moves in the given space and act like global optimizers. Using such algorithms efficiency as well as accuracy is obtained because of their simple and flexible nature.

For the implementation of particle swarm optimization, certain parameters need to be defined. A research space made of particles and the corresponding function, which needs to be optimized. Figure 3 shows the step-by-step process involved in the calculation of the optimal solution using particle swarm optimization. Each particle is associated with certain values, which need to be assigned before the feature selection. The different values are:

- The coordinates of the particle from a given position.
- The speed of the particle so that it can move accordingly with its best neighbour and find a better and optimal position after each iteration.

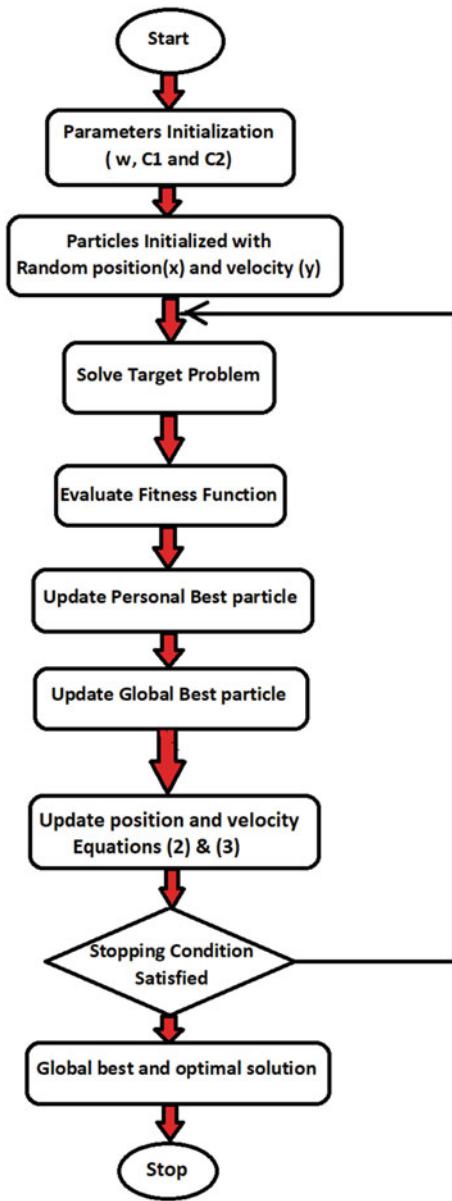


Fig. 3 Particle swarm optimization algorithm process flow chart

- The position of the best neighbour of the swarm so that it can adjust its path of trajectory to find the most optimal path.
- The neighbourhood of the particle, i.e. the particles with which it directly interacts.
- The best-visited position of the particle is close to optimal solution.

3.2.2 Principal Component Analysis

Principal component analysis (PCA) is used for exploration in analytics of data and also in the model in which prediction is taking place. The key feature of PCA is reduction of dimensionality. It is done by the projection of a data point on some of the principal components and to obtain a data, which are dimensionally lower. This is done with the only motive so that analyzing the data becomes easier. Since small datasets are easy to handle, it becomes a lot easier to explore them as well as the computation power for analyzing data becomes a lot faster. However, this is done at the expense of some accuracy. But at the end, it gives a satisfactory result.

For the implementation of principal component analysis, the following steps are undertaken:

- All the continuous initial variables are standardized so that none of them is less contributing than other variables.
- The covariance matrix is computed to check how the variables of the data inputted vary or if any sort of relationship exists between the variables and their mean values.
- When the covariance matrix is formed, the corresponding eigenvectors and eigenvalues are computed so that the principal components can be identified.
- After identifying the principal components we find which components are necessary and which are not. The components with low eigenvalues can be discarded. The resultant matrix formed of vectors is known as Feature Vector.
- In this step, we orient back the data from original axes. We move it to the axes, which are represented by the principal components.

3.3 Classification

Classification plays a huge role in the accuracy of the machine learning model. Better the parameters involved in classification better will be the accuracy. For classification of the data that was subsequently optimized by particle swarm optimization and principal component analysis, hyperparameter tuning [14] was used to achieve the best accuracy by finding the most optimal parameters. Mainly, three classification techniques were used for training of the machine learning model based on their popularity [15] viz support vector machine (SVM), random forest, K-nearest neighbour (KNN). These models were used for each of the dataset. In hyperparameter tuning, four to five attempts were made to classify data in order to find the best parameters that gave satisfying results.

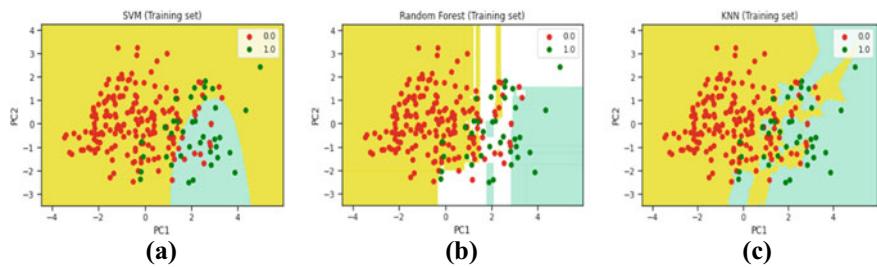


Fig. 4 Scatter plot for training set of heart disease (a) using SVM (b) using random forest (c) k-NN

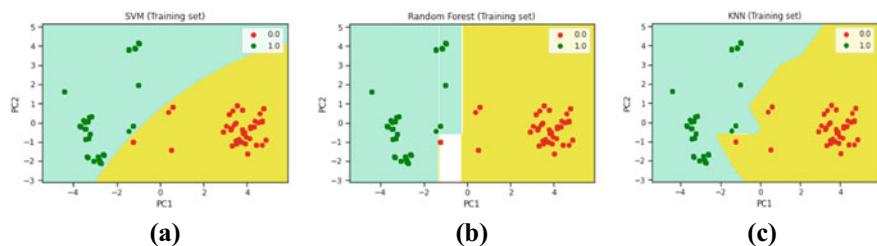


Fig. 5 Scatter plot for training set of cancer (a) using SVM (b) using random forest (c) k-NN

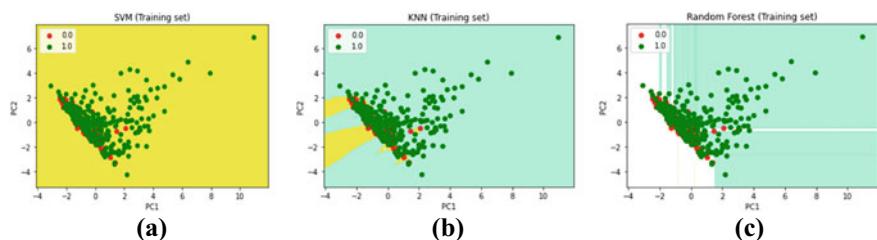


Fig. 6 Scatter plot for training set of liver disease (a) using SVM (b) using random forest (c) k-NN

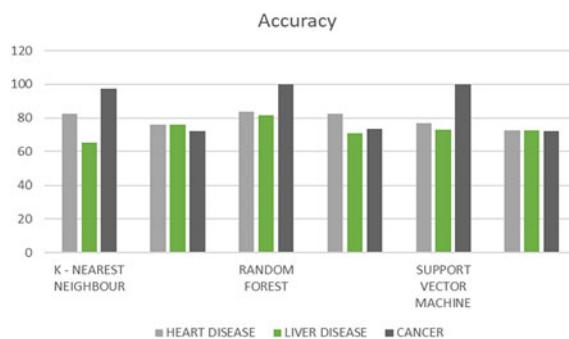
The datasets were trained using various classification techniques and their corresponding scatter plots were obtained in order to show how each algorithm classifies the training sets. Figures 4, 5, 6 show the plots for different classification algorithms used for each disease viz heart disease, cancer and liver disease.

4 Results

Accuracy was calculated for each disease with all the mentioned classifiers. To show the difference of how particle swarm optimization and principal component analysis

Table 1 Accuracy of model using each classifier

Dataset	Accuracy (in %)					
	K-nearest neighbour		Random forest		Support vector machine	
	With PSO and PCA	Without PSO and PCA	With PSO and PCA	Without PSO and PCA	With PSO and PCA	Without PSO and PCA
Heart disease	82.42	75.82	83.52	82.42	76.92	72.50
Liver disease	65.14	75.58	81.32	70.80	73.14	72.50
Cancer	97.40	72.00	100.00	73.20	100.00	72.00

**Fig. 7** Accuracy bar chart

optimizes the result, accuracy was calculated without using them as well as with using them as optimizers. The below table shows the accuracy for heart disease, cardiovascular disease and cancer for each classifier viz K-nearest neighbour (KNN), random forest, support vector machine (SVM). Table 1 shows the classification accuracy for each of the disease with different classification algorithms used for both with particle swarm optimization and principal component analysis and without them. Figure 7 shows the graphical representation of Table 1 in the form of a bar chart to draw more clarity in comparison between the accuracy.

5 Conclusion and Future Scope

The key aim of the proposed system was to optimize the dataset in such a way that the most optimum accuracy can be obtained by selecting only that attributes, which help in categorizing the disease the most. Different classification algorithms were incorporated in this model and then compared. All the data sets taken were preprocessed carefully and then normalized. PSO and PCA were applied to the dataset

for feature optimization. Each classification algorithm was used with hyperparameter tuning. Some showed good results while some lacked accuracy. Results show that the use of hybrid models increases the accuracy of the data sets. This optimized model can be taken as a beginning step in the prediction of various diseases like cardiovascular disease, liver disease and cancer. It can be extended for future research. Hybridization of nature-based optimization techniques can be used. New adaptive learning approaches that are currently being proposed but not developed can be used to improve the accuracy rate of the model.

References

1. K. Kourou, T.P. Exarchos, K.P. Exarchos, M.V. Karamouzis, D.I. Fotiadis, Machine learning applications in cancer prognosis and prediction. *Comput. Struct. Biotechnol. J.* (2015). <https://doi.org/10.1016/j.csbj.2014.11.005>
2. P. S. Optimization, “Particle Swarm Optimization Introduction,” *Optimization*, 2007.
3. H. Abdi, L.J. Williams, Principal component analysis. *Wiley Interdiscip. Rev.: Comput. Stat.* (2010). <https://doi.org/10.1002/wics.101>
4. M.M. Musleh, E. Alajrami, A.J. Khalil, B.S. Abu-Nasser, A.M. Barhoom, S.S. Abu Naser, predicting liver patients using artificial neural network. *Int. J. Acad. Inf. Syst. Res.* **3**(10) (2019)
5. A. Gulia, R. Vohra, P. Rani, Liver patient classification using intelligent techniques (2014)
6. M. Kumari, V. Singh, breast cancer prediction system. *Procedia Comput. Sci.* **132**, 371–376 (2018). <https://doi.org/10.1016/j.procs.2018.05.197>
7. M. Nilashi, O. Ibrahim, H. Ahmadi, L. Shahmoradi, A knowledge-based system for breast cancer classification using fuzzy logic method. *Telemat. Informatics* (2017). <https://doi.org/10.1016/j.tele.2017.01.007>
8. H.E. Hamdaoui, S. Boujraf, N.E.H. Chaoui, M. Maaroufi, A Clinical support system for Prediction of Heart Disease using Machine Learning Techniques, in *2020 5th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)* (2020), pp. 1–5. <https://doi.org/10.1109/ATSIP49331.2020.9231760>.
9. S. Bashir, U. Qamar, M.Y. Javed, An ensemble based decision support framework for intelligent heart disease diagnosis (2015). <https://doi.org/10.1109/i-Society.2014.7009056>
10. R. Detrano et al., International application of a new probability algorithm for the diagnosis of coronary artery disease. *Am. J. Cardiol.* **64**(5), 304–310 (1989). [https://doi.org/10.1016/0002-9149\(89\)90524-9](https://doi.org/10.1016/0002-9149(89)90524-9)
11. “No Title.” <https://www.kaggle.com/rishidamarla/cancer-patients-data>
12. A critical comparative study of liver patients from USA and INDIA: an exploratory analysis. *Int. J. Comput. Sci. Issues* (2012)
13. M. Lichman, UCI machine learning repository [<http://archive.ics.uci.edu/ml>] (2013)
14. R. Bardenet, M. Brendel, B. Kégl, M. Sebag, Collaborative hyperparameter tuning (2013)
15. X. Wu et al., Top 10 algorithms in data mining. *Knowl. Inf. Syst.* (2008). <https://doi.org/10.1007/s10115-007-0114-2>

Bee Intelligence-Guided Partitional Clustering for Outlier Detection



M. Rao Batchanaboyina and Naga Raju Devarakonda

Abstract Parts of data with extreme or uncommon behavior are called outliers. Outlier research is a special area of interest. A variety of techniques have existed in literature. Clustering is one such method to separate outliers from data. However, the existence of outliers itself causes the clustering process complicated. Initialization, position, and the number of cluster centers are key factors for a clustering process. The existing clustering methods are suffering to deal with these issues. Optimization plays a vital role to improve cluster methods. This paper aims to set optimized initial cluster centers to improve clustering results and to separate outlier records through this clustering. Bee colony optimization is embedded into cluster initialization processes. Using this, phase-wise boosted subsets of data from a dataset are processed to optimize cluster centers. Here bee optimization is supposed to improve cluster center initializations. With improved cluster initializations, the partitional clustering process is followed. Outliers are identified through statistical analysis of clustering results. Two legendary datasets from the UCI machine learning repository i.e. Iris data set and Twitter Data set are used to test the proposed methods. The results are encouraging, and the proposed processes are better for clustering and outlier detection that provides finer separation of outliers from data.

Keywords Bee search · Optimized cluster centers · Partition clustering · Cluster statistics · Outlier separation

M. R. Batchanaboyina (✉)

Dept. of CSE, Acharya Nagarjuna University College of Engineering & Technology, Guntur, Andhra Pradesh, India

N. R. Devarakonda

School of Computer Science and Engineering, VIT-AP University, Amaravathi, Andhra Pradesh, India

1 Introduction

Outlier detection is a vital domain of research and has many application areas. Identification of outliers in data provides rich knowledge about data and guides important decision-making with respect to the data context. Information about outliers is actionable in a wide variety of applications such as health diagnosis, information security, strategic planning in business and administration, and so on. Generally, an outlier is a data point that varies greatly from most other data points or that has no predicted behavior.

Outlier detection methods include ways to address the issue of uncovering unusual patterns. The evolving nature of data, differences in data representations, noise in data throw challenges for outlier detection efforts. To address these issues, several methodologies and algorithms have been proposed [1, 2], the improved k-means approach [3, 4] for cluster formation. Nature of input, classes of data, dimensionality, processing time, and accuracy are some of the main issues to deal with in data analysis and outlier detection [5, 6]. The quantity of outliers increases with the size and dimensionality of data [7]. Outlier detection is more challenging in areas like social network analysis [8], trajectory analysis [9], and surveillance analysis systems [10].

Outlier detection approaches can be classified into several subcategories. Some of the popular methods are statistical, distance and density-based, model-based, and cluster-based. Distance-based techniques try to isolate outliers by distance calculations among data points [11, 12]. Density-based techniques search for sparse and dense regions and identify very low-dense regions as outlier spots [13, 14]. Model-based methods construct models through learning where models guide outlier detection [15, 16]. Statistical models try to separate outliers by observing the distribution of data [17]. Clustering methods map data points into homogeneous groups, where poorly grouped points are the candidates for outliers [18]. In this paper, outlier detection through clustering techniques is undertaken to study and develop better means. For getting better cluster optimization consider various optimization techniques like MOWA[19], a popular optimization technique named “Artificial Bee Colony optimization” [20] is studied to design algorithms for better clustering and then to detect outliers in data.

The paper is structured into six sections. In Sect. 2, a comprehensive background on outlier detection using clustering methods and the role of ABC in optimization are given. In Sect. 3, the design of algorithms for clustering and outlier detection is proposed. Section 4 deals with datasets taken for experiments and in Sect. 5 Experimental results are presented along with discussion and finally, conclusions and future scope are made in Sect. 6.

2 Detection of Outlier by Clustering

2.1 *Outlier Detection*

Outlier detection aims to find data points with infrequent behavior. Clustering methods describe the behavior of data by grouping the data points into subsets, which are intra homogeneous and inter heterogeneous. The clusters with an insignificant number of points compared with other clusters are treated as outliers. The worth of a clustering method for outlier detection is dependent on the performance of the method chosen for clustering [21]. Clustering methods are categorized into hierarchical, partitioning, density-based, grid-based, graph-based, and so on. All these methods come under unsupervised learning. So far, a lot of research is taken place in clustering data, which concentrated on minimizing the adverse effects of outliers. Partitional clustering methods based on distance among the data points where the number, initialization of clusters and their centres is aproir task [22]. The hierarchical clustering approach creates a tree-like structure with level-wise cluster formation either in a top-down or bottom-up fashion [23]. Density-based methods require the radius of the neighborhood to model the dense regions, so the number of clusters is not needed in advance. Grid-based clustering partitions the data space into a finite number of cells and the respective densities of these cells guide the clustering [24].

A design of a clustering algorithm for outlier detection keeps the following points for attention: Whether a data point belongs to a particular group or not, and whether a data point outside a cluster can be an outlier or not. What is the distance limit to treat a data point as an outlier? Whether all or part of data points from a sparse cluster can be outliers.

Cluster-based algorithms are a suitable choice for outlier detection in many cases. Being unsupervised and need no prior knowledge of the class, these algorithms are the better choice for stream data. The requirement of knowledge about data distribution is not a constraint for these methods and therefore can allocate new points to a cluster easily. These algorithms are robust to types of data and allow different levels of clustering. Particularly, partitioning-based clustering is easy to implement with reliable outcomes.

Though clustering methods have notable advantages, they have some limitations too. Once an outlier decision is made it never be undone. The decisions on several clusters, outliers, noise in data, and initialization of cluster centers are very sensitive. Data with higher dimensions may become a bottleneck with respect to cost and efficiency.

Regardless of these limitations and drawbacks, cluster-oriented outlier detection is one of the better choices. Clustering-based outlier detection has drawn the attention of researchers as a domain. The decision on cluster width, handling of noise, and the possibility of false positivity are some of the challenges to be addressed for outlier detection through clustering.

All categories of clustering algorithms [25] have their own limitations. The hierarchical clustering process is static [26] and the elements once assigned to a cluster cannot be moved to another cluster. The partitional clustering approach uses fitness criteria based on which a dataset is divided into many groups. This fitness measure guides the cluster formation. In this way, the process tries to minimize distance or maximize intra homogeneity of groups and therefore the partitioning process is converted into the optimization process.

2.2 ABC (*Artificial Bee Colony*) Algorithm

The use of the meta-heuristic approach for optimization is one of the better choices. Optimization tries to minimize cost or maximize profit. An optimization process is built upon some components like objective function f , set of constraints C . An optimization problem is defined as:

Minimize $f(X: x_1, x_2, \dots, x_n)$ subject to the constraints $c_1(X), c_2(X), \dots, c_m(X) \in C$. Here f is an objective function to be minimized, C is a set of constraints on minimization of the objective function and X is the set of decision variables.

The design of optimization algorithms embedding the intelligence of social animals and other creatures got a significant success over the last two decades. Artificial bee colony (ABC) algorithm [27] is one such popular technique and is well known for modesty and strength for optimizing numeric problems. There are numerous variants of this algorithm that covered diverse application areas. The basic method of this algorithm is presented in Fig. 1. The artificial swarm of bees consists of three types of bee functions named employee, onlooker, and scout functions. The role of the employee bee is to search best possible food source (solution). By getting the information from employee bee, the onlooker bee tries to exploit the solution. The role of scout bee is to search a new food source when the nectar of a current food source is exhausted.

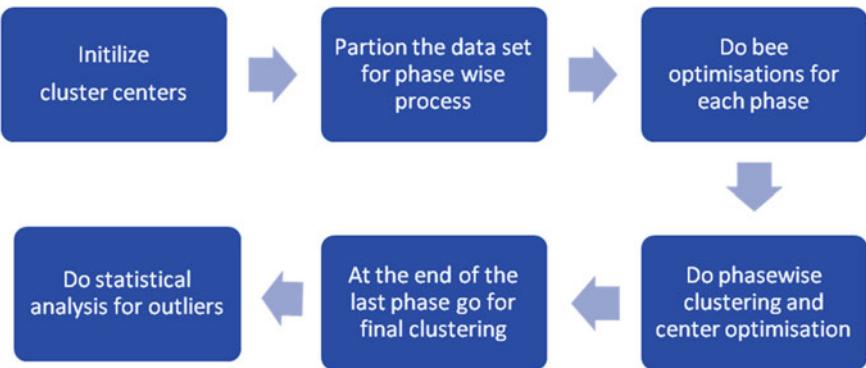


Fig. 1 Methodology overview

ALGORITHM: Basic Artificial Bee Colony (ABC)

1. Do population initializations for food source.
2. Apply employee bees on food sources to evaluate nectar.
3. Compute probabilities for onlooker bees.
4. Apply onlooker bees to choose food sources based on the probabilities.
5. When the nectar of a food source became exhausted, stop the exploitation.
6. Apply the scout bee to search a new food source.
7. Memorise the current best food source.
8. Unless the requirements met, go to step 2.

The functions of ABC are very useful to do optimizations in many application areas. For data clustering also, there is a need of cluster center optimizations in terms of cluster positions and number. If these functions are embedded with clustering process, there is a chance to get better clustering. The usage of ABC optimization in clustering is observed in past research.

A notable problem in clustering through partitional approaches is the initialization of cluster centers. The number and position of cluster centers are also important. The existing methods of clustering are not guaranteed about such initializations and optimizations. In this paper, a new way of optimization is proposed for clustering.

3 Methodology

This methodology aims to identify outliers from data. Clustering in general and partition methods of clustering suffer from outliers. The existing methods have also a problem of local optima where a solution may have convergence to local optimum values instead of a single global optimal solution. Treating these two shortcomings

as a source of thought, the proposed methods tried to find a solution to the problems as well as to find outliers in data.

First, an algorithm is developed to grade a subset of points from a dataset so that the points in the subset can improve cluster centers so that the resultant clustering is improved. To do such grading ABC optimization is embedded in the algorithm where it is appropriate.

Second, one more algorithm is added, which collects the graded data into clusters, does statistical analysis on resultant clusters, and finally picks outliers from the dataset.

Cluster centers initialization

Initialization of cluster centers is a challenging task. A right initialization saves the time and effort for the further process whereas a wrong initialization makes the rest of the process more complex. The differentiation between these two decisions is again a tidy task. In this paper, an initialization based on majority membership is planned.

Algorithm1: Cluster centers Initialization (D, S, k)

D: Data set with N number of records

S: Sample size ($S < N$)

k: Number of cluster centers ($k < S$)

1. Choose S number of sample records.

2. **For** each record r in D

3. **For** each sample record s_i of S

4. Find the minimum distance sample record over the distances $d(s_i, r)$, for $i=1,2,\dots,S$, and allocate the record r to the minimum distance sample group.

5. **End For**

6. **End For**

7. Select top k sample records as initial cluster centers(C), out of S samples based on the descending number of sizes of the sample groups.

Cluster centers optimization

Cluster center optimization is a key process in partition clustering. To do this, the total dataset is portioned into subsets. Each partition is made up with elements that minimize the objective with the initialized centers. The clustering objective function is defined as:

$$\text{OF}(x) = \sum_{i=1}^k d(c_i, x) \quad (1)$$

where d is the distance from a cluster center (c_i) to a record x , k is the number of clusters.

ABC optimization is embedded here to optimize clustering error and hence the optimization function for the bee colony process is the error function.

Algorithm2: Cluster centers optimization (D, C, FN)

D: Data set with N number of records

C: Set of cluster centers

FN: Food number is the size of a partition

1.Select FN number of records randomly from D to form a subset F

2.Apply employee bee search on F to modify F with optimum members.

2.1 **For** each record r in F

2.2 do employ bee search in D to find better replacement x for r, with the objective function OF(x)

2.3 **end For**

3. Apply onlooker bee search on F to modify F further with optimum members.

4.Apply scout bee to locate new record if needed

5. Memorize the best modification for F

Clustering

The next step is to cluster the food records in a partition. This clustering tries to optimize the clustering error. The error function is defined as:

$$\text{Error} = \sum_{i=1}^k \sum_{j=1}^{m_i} d(c_i, x_j) \quad (2)$$

where d is the distance from a cluster center (c_i) to its elements (x_j s), k represents the number of clusters and m_i represents the number of members of cluster i .

Algorithm3: Clustering on food records (F, C)

C: Set of cluster centers

F: Partition subset

1.**For** each record r in F3.**For** each partition food record f_i of F4. Find the minimum distance food record over the distances $d(f_{i,r})$, for $i=1, 2, \dots, k$, and allocate the record r to the minimum distance cluster.5.**End For**6.**End For**

7.find the averages of each cluster and modify the centers set C.

8.reallocate the elements to the clusters.

8.find the clustering error

9.if the error term converges stop

10. else go to step 1

The proposed process of clustering is a mix of partition clustering and center optimization by bee intelligence. To do so, a dataset is supposed to be partitioned into smaller units where each unit is supposed to be processed in phases. The first phase starts with cluster initialization. Based on the initial cluster centers, the first unit of data is processed. The process tries to check each record of the unit for center optimizations. The total distance of record from all cluster centers is calculated and the calculated value is treated as the current solution. Bee intelligence in terms of employee and onlooker bee functions tries to improve the current solution by replacing the records of the current unit with better records. A current solution can be improved if a record is replaceable by another record whose total distance from all centers is less than the current solution. At the end of the first phase, the first unit of the dataset has data records that are most suitable to current cluster centers. Now clustering is done for this unit to get updated and optimized centers. The same process is continued until the total dataset is clustered. In this way, one can get better clustering results. Clusters with the majority of outliers can be separable now. The result of clustering now undergoes further outlier detection process through statistical analysis. The diagrammatic representation of the process is presented in Fig. 1.

The summary of the proposed clustering process is given below:

The total process uses N number of records, which are going to subset into P partitions where the size of each partition is N/P .

Algorithm: Bee intelligence for Outlier Detection

1. Call **Cluster centers Initialization** (D, S, k) (**Algorithm1**)
 2. Choose the number of partitions P
 3. **For** each partition number p=1, 2, 3,.....P
 4. FN=P*N/P
 5. Call **Cluster centers optimization** (D, C, FN) (**Algorithm2**)
 6. Call **Clustering on food records** (F, C) (**Algorithm3**)
 7. End **For**
-

Outlier identification

After the clustering process is completed, the next step is to identify outliers in data. A cluster that behaves differently from normal behavior is supposed to be an outlier cluster. This can be identified by summarizing the attributed values of a cluster like a cluster size, mean values of attributes, etc. Besides, different statistical techniques are identified to pick outliers in a cluster.

- (A) **$\mu-\sigma$ method:** Outlier detection with limitations on the mean(μ) and standard deviation(σ) of the cluster elements is proposed. Here a cluster element that falls outside ($\mu - 3\sigma, \mu + 3\sigma$) is supposed to be an outlier.
- (B) **Quartile limits method:** Outlier detection with quartile values of the cluster elements is proposed. Here a cluster element falls outside (first quartile – 1.5*interquartile range, third quartile + 1.5*interquartile range) is supposed to be an outlier.
- (C) **Mini-Max limit method:** Here, outlier identification is based on one of the dataset's attribute features.
Maximum and minimum values of that attributes identify outliers using the following formulae:
 - (i) If $(\text{Max}(a) - \text{min}(a))/2 < \text{current value } a$, then the current record is an outlier where **a** is an attribute.
 - (ii) If $\text{Average}(a) < \text{current value } a$, then the current record is an outlier where **a** is an attribute.
- (D) **Best β method:** A record is named as best record β for a cluster when it is nearest to the cluster center. Based on best position, other records are classified as follows:

A record in a cluster that falls outside the β -neighborhood ($\beta - 3\sigma, \beta + 3\sigma$) is supposed to treat as outlier.

4 Data Sets

Twitter Dataset:

A social network dataset that is a compilation of instances of buzz events is a Twitter data set. Twitter data set is aimed to identify the magnitude of buzz in social media. The values of 77 attributes are aggregated and normalized to get a dataset consists of 11 actual features with normalized values. The class of each record is also available, which is used to compare the results of proposed clustering and outlier detection. There are 27,775 outliers in the data. These are called buzz records. The rest of the records are normal records. Therefore, it is two-class data (Normal/Buzz). The proposed methodologies are applied to this dataset to separate normal and buzz records through clustering.

Iris dataset:

Iris dataset has 150 observations equally distributed observations among the three species—Setosa, Versicolor and Virginica. It is a labeled dataset and so the dataset is used without the *Species* column. After data clustering, it is possible to compare the predicted results with the original results (labels), to the accuracy of the model.

5 Results and Discussion

Clustering results and outlier identification with Twitter dataset

Clustering results along with outlier information are presented in Fig. 2. There are 20 clusters formed. Clusters numbered 4, 5, 9, 13 and 18 with a total of 21921 records are identified as buzz clusters as they are having more than 90% of buzz records. Clusters numbered 11, 14, and 19 have moderate amounts of outlier records, which are totaled to the number 4250. The rest of the 12 clusters almost have normal records. This classification reveals that the proposed approach can separate outlier clusters representing above 95% of the outliers. The existing partition clustering methods are

Fig. 2 Outlier information

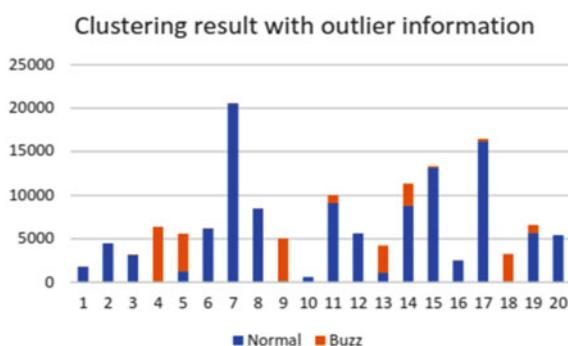


Table 1 Iris data clustering by k-means

Cluster versus class	Iris-setosa	Iris-versicolor	Iris-virginica
1	0	48	14
2	0	2	36
3	50	0	0

Table 2 Iris data clustering by the proposed algorithm

Cluster versus class	Iris-setosa	Iris-versicolor	Iris-virginica
1	0	42	04
2	0	8	46
3	50	0	0

compared with this proposed method and the outlier separation percentage of such method is below 90%.

Clustering results of Iris dataset

Firstly k -means algorithm is applied to the dataset. The results are shown in Table 1. The highlighted values represent a true class.

Clustering accuracy:

Clustering accuracy can be defined as the ratio of the sum of correctly classified records in each class to total records. This is given by the following equation. Therefore,

$$\text{Cluster accuracy} = (\text{Truly classified records}/\text{Total records}) \times 100$$

The calculated accuracy = 89.33% (3)

Now the proposed algorithm is applied to the dataset. The results are shown in Table 2. The highlighted values represent a true class.

$$\text{Cluster accuracy (for proposed method)} = 92\%$$

The comparison

Comparison between existing and proposed approaches with respect to statistical methods of outlier identification:

The proposed method of clustering is compared with a well-known existing method for clustering (k -means) and the results are compared. The comparison is presented in the table. In the table, the values represent the percentages of matched outliers when they are identified with the statistical measures named in the table columns. From the table, it can be observed that the proposed methods of outlier detection are better than existing methods.

Method/technique (outlier matching %)	$\mu-\sigma$ method	Quartile limits method	Mini–max limit method	Best β method
Existing	93	66	99	75
Proposed	90	59	87	70

Comparison of results for Iris dataset

Algorithm	Accuracy (%)
Existing	89.33
Proposed	92

After observing these accuracy counts, it can be observed that the proposed process can get more accurate clustering.

Discussion

These two experiments provided results that show the ability of the proposed method to do better clustering. The accuracy of the clustering depends on the correct position and number of cluster centers. The proposed processes can do such optimizations. Artificial bee functions played a key role in cluster center optimization. Results of clustering provide a way to separate outliers. Further statistical analysis of clustering results helped to separate outlier records further. However, the influence of outliers in data still misleads the clustering process. Noise or extreme values of some attributes cause outliers. If such attributes are identified and separated, the influence of outliers can be managed. The methods proposed here tried to do such separations.

6 Conclusion and Future Scope

Outliers in data influence data analysis. The separation of such outliers is a challenge. Clustering is a way to deal with outliers. In this paper, partitional clustering is undertaken. To get better partition centers, an optimization technique based on bee intelligence is proposed. A phase-wise data clustering is followed in which, cluster centers are optimized so that the result of clustering is more accurate. The proposed models are being validated using data sets from the UCI machine learning repository. Results of clustering using existing partitional clustering (k -means) and the proposed clustering are compared. Statistical-based outlier identification methods are used to match the results. The proposed models of clustering and outlier identification provided better results of clustering and outlier detection. Statistical study on clustering can give better insights to identify outliers. These methods certainly show the ways for further research in outlier analysis.

References

1. V. Hodge, J. Austin, A survey of outlier detection methodologies. *Artif. Intell. Rev.* **22**(2), 85–126 (2004)
2. Z. Niu, S. Shi, J. Sun, X. He, A survey of outlier detection methodologies and their applications, in *International Conference on Artificial Intelligence and Computational Intelligence* (Springer, Berlin, Heidelberg, 2011, September), pp. 380–387
3. S.S. Ramakrishna, T. Anuradha, An effective framework for data clustering using improved k-means approACH. *Int. J. Adv. Res. Comput. Sci.* **9**(2) (2018)
4. M.R. Batchanaboyina, N. Devarakonda, An effective approach for selecting cluster centroids for the k-means algorithm using IABC approach, in *2019 IEEE 18th Int'l Conf. on Cognitive Informatics & Cognitive Computing (ICCI*CC'19)*
5. C.C. Aggarwal, S.Y. Philip, An effective and efficient algorithm for high-dimensional outlier detection. *VLDB J.* **14**(2), 211–221 (2005)
6. M.R. Batchanaboyina, N. Devarakonda, Design and evaluation of outlier detection based on semantic condensed nearest neighbor. *J. Intell. Syst.* (2019) aop
7. H.P. Kriegel, P. Kröger, E. Schubert, A. Zimek, Outlier detection in axis-parallel subspaces of high dimensional data, in *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (Springer, Berlin, Heidelberg, 2009, April), pp. 831–838
8. Z. Cai, Z. He, X. Guan, Y. Li, ‘Collective data-sanitization for preventing sensitive information inference attacks in social networks. *IEEE Trans. Depend. Sec. Comput.* **15**(4), 577–590 (2018)
9. Y. Yu, L. Cao, E.A. Rundensteiner, Q. Wang, Outlier detection over massive-scale trajectory streams. *ACM Trans. Database Syst.* **42**(2), 10:1–10:33 (2017)
10. T. Xiao, C. Zhang, H. Zha, Learning to detect anomalies in surveillance video. *IEEE Signal Process. Lett.* **22**(9), 1477–1481 (2015)
11. E.M. Knox, R.T. Ng, Algorithms for mining distance based outliers in large datasets, in *Proceedings of the International Conference on Very Large Data Bases* (1998, August, Citeseer), pp. 392–403
12. T.T. Dang, H.Y.T. Ngan, W. Liu, Distance-based k-nearest neighbours outlier detection method in large-scale traffic data, in *Proc. IEEE Int. Conf. Digital Signal Process.* (Jul. 2015), pp. 507–510
13. F. Keller, E. Müller, K. Bohm, HiCS: high contrast subspaces for density-based outlier ranking, in *2012 IEEE 28th International Conference on Data Engineering* (2012, April, IEEE), pp. 1037–1048
14. M. Bai, X. Wang, J. Xin, G. Wang, An efficient algorithm for distributed density-based outlier detection on big data. *Neurocomputing* **181**, 19–28 (2016)
15. C. Fraley, A.E. Raftery, Model-based clustering, discriminant analysis, and density estimation. *J. Am. Stat. Assoc.* **97**(458), 611–631 (2002)
16. P. Sharma, S. Sundaram, Diagnosis of Parkinson’s disease using modified grey wolf optimization. *Cogn. Syst. Res.* **54**(2019), 100–115 (2018)
17. P.J. Rousseeuw, M. Hubert, Robust statistics for outlier detection. *Wiley Interdiscip. Rev.: Data Min. Knowl. Discov.* **1**(1), 73–79 (2011)
18. L. Duan, L. Xu, Y. Liu, J. Lee, Cluster-based outlier detection. *Ann. Oper. Res.* **168**(1), 151–168 (2009)
19. R. Jain, D. Gupta, A. Khanna, Usability feature optimization using MWOA, in *International Conference on Innovative Computing and Communications, Lecture Notes in Networks and Systems*, vol. 56 (2019). https://doi.org/10.1007/978-981-13-2354-6_47
20. M.R. Batchnaboyina, N.R. Devarakonda, Handling optimization problem, and the scope of varied artificial bee colony (ABC) algorithms: a contemporary research. *IJITEE* **8**(6S4) (2019). ISSN: 2278-3075
21. C. Fan, F. Xiao, Z. Li, J. Wang, Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: a review. *Energy Build.* **159**, 296–308 (2018)

22. A. Dharmarajan, T. Velmurugan, Applications of partition based clustering algorithms: a survey, in *2013 IEEE International Conference on Computational Intelligence and Computing Research* (IEEE, 2013, December), pp. 1–5
23. A. Szymkowiak, J. Larsen, L.K. Hansen, Hierarchical clustering for datamining, in *Proceedings of KES-2001 Fifth International Conference on Knowledge-Based Intelligent Information Engineering Systems & Allied Technologies* (2001, June), pp. 261–265
24. B. Wu, B.M. Wilamowski, A fast density and grid based clustering method for data with arbitrary shapes and noise. *IEEE Trans. Industr. Inf.* **13**(4), 1620–1628 (2016)
25. S. Das, A. Abraham, A. Konar, *Metaheuristic Clustering*, vol. 178 (Springer)
26. A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review. *ACM computing surveys (CSUR)* **31**(3), 264–323 (1999)
27. D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm. *Appl. Soft Comput.* **8**(1), 687–697 (2008)

Author Index

A

- Abraham, Shilpa Elsa, 699
Abul Kashem, Mohammad, 563
Agrawal, Sarthak, 435
Ahmad, Waqar, 345
Ahuja, Devansh, 609
Akter, Laboni, 191
Akter, Tamanna, 223
Altaf, Ifra, 785
Alvi, Nasif, 41, 77, 191, 223
Amutha, A. L., 635
Anand, Abhineet, 379
Anjali, T., 573
Anuranjana, 645
Arora, Mehardeep Singh, 23
Arora, Monika, 323
Arora, Praveen, 215
Arora, Sakshi, 179
Asharaf, S., 285
Aubi, Ishmeet Kaur, 531
Augustine, Jisha, 777

B

- Bagla, Nikhil, 53
Bahrani, Anshul, 609
Bansal, Poonam, 745
Bartere, Mahip M., 733
Batchanaboyina, M. Rao, 813
Battineni, Gopi, 345
Bhan, Nabadh, 757
Bhatia, Manjot Kaur, 475
Bhimarapu, Akshita, 263
Bide, Pramod, 415
Biilah, Md. Al-Mamun, 223

Bishnoi, Sangeeta, 553

- Bristy, Nusrat Jahan, 223
Brunet, R. Golda, 635
Butt, Muheet Ahmed, 593, 785

C

- Chand, Satish, 623
Chaturvedi, Ruchi, 263
Chauhan, Ekansh, 53, 801
Chauhan, Swati, 531
Chugh, Aarti, 475
Churi, Prathamesh, 445

D

- Deen Muhammad, Sumaiya, 167
Deshpande, Santosh L., 205
Devarakonda, Naga Raju, 813
Dhingra, Bhumika, 323
Diddee, Harshita, 253
Dutta, Prateek, 263
Dwivedi, A. K., 117

E

- Eashwaran, R., 315

F

- Fayaz, Sheikh Amir, 593
Ferdib-Al-Islam, 191

G

- Garg, Nitika, 435

Ghansiyal, Adrija, 623
 Ghosh, Mounita, 191
 Goel, Gaurav, 379
 Goel, Navansh, 355
 Goel, Varun, 801
 Gopalakrishnan, Sruthi, 129
 Gopan, Neethu Radha, 129
 Goyal, Muskan, 461
 Gupta, Amit Kumar, 541
 Gupta, Deepak, 53, 253, 461
 Gupta, Dhruv, 323
 Gupta, Dr. Deepak, 335
 Gupta, Pankaj Kumar, 585

H

Hossain, Iqbal, 563
 Hossain, Md. Mobarak, 563

I

Islam, Tanvir, 665

J

Jadhav, Dipti, 655, 683
 Jain, Alok, 335
 Jain, Charu, 475
 Jain, Prabhav, 801
 Jain, Rocky, 745
 Janghel, Rekh Ram, 517
 Jennath, H. S., 285
 Jereesh, A. S., 777

K

Kaliwal, Rohit B., 205
 Kalra, Mala, 275
 Kansra, Bhrigu, 253
 Kapoor, Bhaskar, 53
 Karsi, Priya, 609
 Kaur, Preeti, 87
 Kaur, Sanmukh, 645, 757
 Kayande, Vedant, 239
 Khanna, Ashish, 253, 485
 Khanna, Pooja, 497
 Khedkar, Sujata, 609
 Kovoor, Binsu C., 699
 Kumar, Bijender, 623
 Kumari, Anshu, 391
 Kumari, Pulak, 541
 Kumar, Manoj, 709, 769
 Kumar, Nitin, 345
 Kumar, Rajesh, 141

Kumar, Rakesh, 275
 Kumar, Sachin, 497
 Kumar, Sharlin, 69
 Kumar, Sumit, 379
 Kumar, Tejaswi, 355
 Kumar Yadav, Dharmendra, 769
 Kundale, Jyoti, 683

L

Lokulwar, Prasad P., 733
 Lubana, Anurupa, 645

M

Mahato, Ashish Raj, 403
 Mahendiram, Sruthi, 129
 Mahmud, Ikbal, 41
 Majumder, Agnideepa, 23
 Malik, Shaily, 745
 Mandal, Riddhi, 263
 Manoj, E., 573
 Mantri, Palak, 23
 Mehta, Kamakshi, 585
 Mishra, Piyush, 239
 Mittal, Mamta, 345
 Mittal, Nimisha, 461
 Mittal, Prabhat, 585
 Mittal, Shubham, 335
 Mittal, Tanya, 315
 Mridha, M. F., 665

N

Nema, Anshula, 623

O

Oruganti, Ramakrishna, 445
 Oswald, C., 355

P

Pagad, Naveen S., 151
 Pai, Pratik, 239
 Paithankar, Ketan, 263
 Panda, Supriya P., 365
 Pandey, Pallavi, 509
 Pandey, Prerna, 721
 Parikh, Abhishek, 11
 Parmar, Manish, 239
 Patel, Bhargav, 11
 Patel, Hiral A., 11
 Patel, Shaswat, 87

- Patil, Gaurav, 263
 Paulson, Rosalind Margaret, 129
 Pradeep, N., 151
 Pragya, 497
 Prakash, Ujjawal, 391
 Prasad, Sanjeev Kumar, 531, 541
 Priyadarshi, Anurag, 541
 Purohit, Rajendra, 553
- R**
 Raghuvanshi, Naman, 69
 Raihan, M., 191, 223
 Raihan, Md. Mohsin Sarker, 191
 Rani, Kumud, 275
 Rani, Poonam, 69
 Ranjan, Rakesh, 1
 Rao, A. Prabhakara, 1
 Rashid, Shazia, 299
 Rathie, Dishant, 709
 Rathor, Sandeep, 435
 Rehana, Hasin, 223
 Rodrigues, Joel J. P. C., 253
 Roselyn, J. Preetha, 635
 Roy, Diti, 41
 Roy, Siddhant, 355
 Roy, Tamal Joyti, 41
- S**
 Sahoo, Sushmita, 403
 Sakarkar, Gopal, 263
 Sandesara, Harsh, 415
 Santoshi, Seneha, 299
 Saraf, Pranay D., 733
 Saraswat, Nikki, 721
 Saxena, Akash, 117, 141
 Saxena, Ankur, 23
 Saxena, Vishal, 117
 Seeja, K. R., 509
 Shah, Binil, 87
 Shah, Karan, 415
 Sharma, Ajay, 141
 Sharma, Ashish, 427
 Sharma, Chakshu, 485
 Sharma, Harish, 141
 Sharma, Kavita, 721
 Sharma, Madhav, 391
 Sharma, Pratham, 745
 Sharma, Tanya, 299
 Sharma, Vivek Kumar, 475
 Sharmin, Sanjida, 167
 Sheikh, Tariq Hussain, 253
- Shekhawat, Shalini, 117, 141
 Sheoran, Kavita, 315
 Shinde, Amol, 655
 Shinde, Swapnil, 655
 Shivam, 263
 Shivhare, Shiv Naresh, 721
 Shokeen, Jyoti, 69
 Shukla, Priyansh, 721
 Singh, Anmol, 69
 Singh, Anshul, 69
 Singh, Bhawan Deep, 485
 Singh, Dajinder, 323
 Singh, Drishti, 573
 Singh, Pradhuman, 239
 Singh, Richa, 53
 Singh, Simarjeet, 517
 Singla, Kanika, 391, 403
 Singla, Ramendra, 335
 Sirswal, Manpreet, 53
 Srambical, Varghese Paul, 129
 Srivastava, Yashi, 497
 Sukhi, Shamima, 665
 Suri, Bhawna, 215
 Swamy, Sowmya, 683
- T**
 Talukder, Kamrul Hasan, 77
 Tandon, J. K., 585
 Taneja, Shweta, 215
 Tanwar, Soumya, 215
 Tasnim, Farzana, 167
 Tiwari, Rajeev, 379
 Tomer, Minakshi, 709
 Tripathi, Divya, 101
 Tripathi, Jyoti, 623
- U**
 Uddin, Abdul Hasib, 77
 Upreti, Ravi, 345
 Usama Islam, Muhammad, 563
 Uthra, R. Annie, 635
- V**
 Vashisht, Ankit, 745
 Verma, Prateek, 435
 Vij, Richa, 179
 Vishwakarma, Anish Kumar, 1
- W**
 Waheed, Abdul, 461

Wairyta, Subodh, [101](#)

Wilson, Rakshit Luke, [403](#)

Z

Zaman, Majid, [593, 785](#)

Zaman, Mohammad Nawai, [299](#)

Zinat, Sakila Mahbin, [665](#)

Y

Yadav, Neelam, [365](#)