

Harish Sharma  
Antorweep Chakravorty  
Shahid Hussain  
Rajani Kumari *Editors*

# Artificial Intelligence: Theory and Applications

Proceedings of AITA 2023, Volume 2

# **Lecture Notes in Networks and Systems**

**Volume 844**

## **Series Editor**

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

## **Advisory Editors**

Fernando Gomide, Department of Computer Engineering and Automation—DCA, School of Electrical and Computer Engineering—FEEC, University of Campinas—UNICAMP, São Paulo, Brazil

Okyay Kaynak, Department of Electrical and Electronic Engineering, Bogazici University, Istanbul, Türkiye

Derong Liu, Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, USA

Institute of Automation, Chinese Academy of Sciences, Beijing, China

Witold Pedrycz, Department of Electrical and Computer Engineering, University of Alberta, Alberta, Canada

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

Marios M. Polycarpou, Department of Electrical and Computer Engineering, KIOS Research Center for Intelligent Systems and Networks, University of Cyprus, Nicosia, Cyprus

Imre J. Rudas, Óbuda University, Budapest, Hungary

Jun Wang, Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong

The series “Lecture Notes in Networks and Systems” publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

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

For proposals from Asia please contact Aninda Bose ([aninda.bose@springer.com](mailto:aninda.bose@springer.com)).

Harish Sharma · Antorweep Chakravorty ·  
Shahid Hussain · Rajani Kumari  
Editors

# Artificial Intelligence: Theory and Applications

Proceedings of AITA 2023, Volume 2



Springer

*Editors*

Harish Sharma  
Department of Computer Science  
and Engineering  
Rajasthan Technical University  
Kota, Rajasthan, India

Shahid Hussain  
Biomedical Robotics, Information,  
Technology (IT) and Systems  
University of Canberra  
Bruce, ACT, Australia

Antorweep Chakravorty  
Department of Computer Science  
and Electrical Engineering  
University of Stavanger  
Stavanger, Norway

Rajani Kumari  
IBS, Bangalore, Off-Campus Centre  
of ICFAI Foundation for Higher Education  
(IFHE) University  
Bengaluru, Karnataka, India

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-99-8478-7

ISBN 978-981-99-8479-4 (eBook)

<https://doi.org/10.1007/978-981-99-8479-4>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024

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

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

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

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

Paper in this product is recyclable.

# Preface

This book contains outstanding research papers as the proceedings of the International Conference on Artificial Intelligence: Theory and Applications (AITA 2023). AITA 2023 has been organized by ICFAI Business School (IBS), Bangalore, India, and technically sponsored by Soft Computing Research Society, India. The conference is conceived as a platform for disseminating and exchanging ideas, concepts, and results of researchers from academia and industry to develop a comprehensive understanding of the challenges of the advancements of artificial intelligence and its applications to solve complex problems. This book will help in strengthening congenial networking between academia and industry. We have tried our best to enrich the quality of the AITA 2023 through the stringent and careful peer-review process. This book presents novel contributions to artificial intelligence and serves as reference material for advanced research.

We have tried our best to enrich the quality of the AITA 2023 through a stringent and careful peer-review process. AITA 2023 received many technical contributed articles from distinguished participants from home and abroad. AITA 2023 received 648 research submissions from different countries. After a very stringent peer-reviewing process, only 78 high-quality papers were finally accepted for presentation and the final proceedings.

This book presents first volume of 38 research papers of data science and applications and serves as reference material for advanced research.

Kota, India  
Stavanger, Norway  
Canberra, Australia  
Bengaluru, India

Harish Sharma  
Antorweep Chakravorty  
Shahid Hussain  
Rajani Kumari

# Contents

<b>Spam Email Image Detection Using Convolution Neural Network and Convolutional Block Attention Module .....</b>	1
Md Asif Jamal and Pradeep Kumar	
<b>Identifying Fake Twitter Trends with Deep Learning .....</b>	15
Thahab M. AlBuhairi and Haya A. Alhakbani	
<b>An Intelligent Agent Framework for Resilient Deployment in the Internet of Things Environment .....</b>	29
Nainsi Soni and Saurabh Kumar	
<b>The Impact of Financial Ratios and Pandemic on Firm Performance: An Indian Economic Study .....</b>	41
Manpreet Kaur Khurana, Shweta Sharma, and Navneet Bhargava	
<b>An Enhanced BERT Model for Depression Detection on Social Media Posts .....</b>	53
R. Nareshkumar and K. Nimala	
<b>Quality Prediction of a Stack Overflow Question Using Machine Learning .....</b>	65
Tanvi Mehta, Samruddhi Multaikar, Srushti Patil, and Namrata Gawande	
<b>Decision-Making Framework for Supplier Selection Using an Integrated Approach of Dempster-Shafer Theory and Maximum Entropy Principle .....</b>	81
Garima Bisht and A. K. Pal	
<b>Improved Accuracy of Robotic Arm Using Virtual Environment .....</b>	95
Utkarsh Rastogi, Javed Sayyad, B. T. Ramesh, and Arunkumar Bongale	
<b>Human Activity Recognition a Comparison Between Residual Neural Network and Recurrent Neural Network .....</b>	109
K. P. Anu and J. V. Bibal Benifa	

<b>Using AI Planning to Automate Cloud Infrastructure .....</b>	<b>125</b>
Vijay Prakash, Leonardo Freitas, Lalit Garg, and Pardeep Singh	
<b>Using Historical Trip Information to Determine the Waiting Time Required for Taxi Services .....</b>	<b>139</b>
Michael Vassallo, Vijay Prakash, Lalit Garg, and Pardeep Singh	
<b>The Impact of Cesarean Section Trends and Associated Complications in the Current World: A Comprehensive Analysis</b>	
<b>Using Machine Learning Techniques .....</b>	<b>153</b>
K. Mallikharjuna Rao, Harleen Kaur, and Sanjam Kaur Bedi	
<b>Novel Hybrid Methods for Journal Article Summarization Combining Graph Method and Rough Set TFIDF Method with Pegasus Model .....</b>	<b>173</b>
K. Sheena Kurian and Sheena Mathew	
<b>Differential Evolution Wrapper-Based Feature Selection Method for Stroke Prediction .....</b>	<b>191</b>
Santwana Gudadhe and Anuradha Thakare	
<b>Parkinson's Disease Identification from Speech Signals Using Machine Learning Models .....</b>	<b>201</b>
Rahul Saxena and J. Andrew	
<b>Performance Analysis of Deep CNN, YOLO, and LeNet for Handwritten Digit Classification .....</b>	<b>215</b>
Jibok Sarmah, Madan Lal Saini, Ankush Kumar, and Vidhan Chasta	
<b>Data-Driven Decision Support Systems in E-Governance: Leveraging AI for Policymaking .....</b>	<b>229</b>
Anudeep Arora, Prashant Vats, Neha Tomer, Ranjeeta Kaur, Ashok Kumar Saini, Sayar Singh Shekhawat, and Monika Roopak	
<b>The Infrastructure Development of Contemporary Medical Devices Based on Internet of Things Technology .....</b>	<b>245</b>
Haider Al-Kanan and Ahmed S. Alzuhairi	
<b>Business Intelligence System Adoption Project in the Area of Investments in Financial Assets .....</b>	<b>259</b>
Beata Dratwińska-Kania and Aleksandra Ferens	
<b>Feature Selection Techniques to Enhance Prediction of Clinical Appointment No-Shows Using Neural Network .....</b>	<b>275</b>
Jeffin Joseph, S. Senith, A. Alfred Kirubaraj, and S. R. Jino Ramson	
<b>A Simple Recommendation Model Using the Item's Global Popularity and Frequency-Based User Preference .....</b>	<b>287</b>
Somaraju Suvvari and Md Iftekhar Ahmad	

<b>An Early Detection of Autism Spectrum Disorder Using PDNN and ABIDE I&amp;II Dataset .....</b>	295
Manjunath Ramanna Lamani and P. Julian Benadit	
<b>Comparison of Machine Learning-Based Intrusion Detection Systems Using UNSW-NB15 Dataset .....</b>	311
Rakoth Kandan Sambandam, D. Daniel, R. Gokulapriya, Divya Vetriveeran, J. Jenefa, and Anuneshwar	
<b>Comparative Analysis of Face Recognition Models for Criminal Detection .....</b>	325
Gouri Goyal, Vaibhav Kumar, Piyush Aggarwal, Gunjan Chugh, and Tripti Lamba	
<b>Transcription of Ancient Indian Manuscripts Through Artificial Intelligence—Current Status of Technology and the Way Forward .....</b>	339
R. Harish and G. N. Raghavendra Rao	
<b>Computer Vision and Convolutional Neural Network for Dense Crowd Count Detection .....</b>	353
D. Sirisha, S. Sambhu Prasad, and Subodh Kumar	
<b>Evaluation Methods of CDIO Project at Duy Tan University .....</b>	363
Van-Truong Truong and Anand Nayyar	
<b>Deciphering Stem Cell Pluripotency Using a Machine Learning Clustering Approach .....</b>	375
Nikhil Jain, Payal Gupta, Abhishek Sengupta, Ankur Chaurasia, and Priyanka Narad	
<b>Extremal Trees of the Reformulated and the Entire Zagreb Indices .....</b>	389
Anjusha Asok and Joseph Varghese Kureethara	
<b>Sign Language Interpreter Using Stacked LSTM-GRU .....</b>	405
M. Dhilsath Fathima, R. Hariharan, Sachi Shome, Manbha Kharsiyemlieh, J. Deepa, and K. Jayanthi	
<b>Can Learning Games Facilitate Open Innovation Capacity in IT Industry? The Case of Resilience .....</b>	417
Eleni G. Makri	
<b>Investigating Role of SVM, Decision Tree, KNN, ANN in Classification of Diabetic Patient Dataset .....</b>	431
Sarita Kumari and Amrita Upadhyaya	
<b>Optimal Resource Allocation in Cloud Computing Using Novel ACO-DE Algorithm .....</b>	443
Himanshu Bhusan Sahoo and D. Chandrasekhar Rao	

<b>Prediction of Cardiovascular Disease by Feature Selection and Machine Learning Techniques .....</b>	<b>457</b>
Aditya Ranade and Nitin Pise	
<b>Performance Evaluation and Comparative Analysis of Machine Learning Techniques to Predict the Chronic Kidney Disease .....</b>	<b>473</b>
Majid Bashir Malik, Mohd Ali, Sadiya Bashir, and Shahid Mohammad Ganie	
<b>Designer Face Mask Detection Using Marker-Based Watershed Transform and YOLOv2 CNN Model .....</b>	<b>487</b>
Arpita Vyas and Jankiballabh Sharma	
<b>Drought Prediction Using Machine Learning Forecasting Model in the Context of Bangladesh During 1981–2018 .....</b>	<b>499</b>
Alomgir Hossain, Momotaz Begum, and Nasim Akhtar	
<b>A Survey on Various Aspects of Recommendation System Based on Sentiment Analysis .....</b>	<b>517</b>
Rohit Mittal, Sumit Kumar, Vishal Shrivastava, Vibhakar Pathak, and G. L. Saini	
<b>Author Index .....</b>	<b>531</b>

# Editors and Contributors

## About the Editors

**Harish Sharma** is an associate professor at Rajasthan Technical University, Kota, in Department of Computer Science & Engineering. He has worked at Vardhaman Mahaveer Open University, Kota, and Government Engineering College Jhalawar. He received his B.Tech. and M.Tech. degrees in Computer Engineering from Government Engineering College, Kota, and Rajasthan Technical University, Kota, in 2003 and 2009, respectively. He obtained his Ph.D. from ABV-Indian Institute of Information Technology and Management, Gwalior, India. He is the secretary and one of the founder members of Soft Computing Research Society of India. He is a lifetime member of Cryptology Research Society of India, ISI, Kolkata. He is an associate editor of *International Journal of Swarm Intelligence (IJSI)* published by Inder-science. He has also edited special issues of the many reputed journals like *Memetic Computing*, *Journal of Experimental and Theoretical Artificial Intelligence*, *Evolutionary Intelligence*, etc. His primary area of interest is nature inspired optimization techniques. He has contributed in more than 105 papers published in various international journals and conferences.

**Dr. Antorweep Chakravorty** is an associate professor at the University of Stavanger. His current research and development work is in the field of applied Blockchains, Big Data, Large-Scale Machine Learning, and Data Privacy. He has an interest in real-world problems, especially development of privacy enabled data-driven services in smart energy, health care, and smart city domains. Antorweep completed his Ph.D. in 2015 with a thesis on Privacy Preserving Big Data Analytics at the University of Stavanger, Norway. Along with having a background in applied research in data-driven solutions, he is also involved in mentoring, teaching, and supervision.

**Dr. Shahid Hussain** is working at University of Canberra as an associate professor of Biomedical Robotics. Prior to that, he has worked as a lecturer at University of

Wollongong, Australia. Dr. Hussain has obtained his Ph.D. in Mechanical Engineering from the University of Auckland, New Zealand, in 2013. His research interests include assistive and rehabilitation robotics, compliant actuation of robots, robot mechanism design and optimization, nonlinear dynamics and control of robotic systems, human–robot interaction, biomechanical modeling, engineering education, and micro-electro-mechanical systems (MEMS). Dr. Hussain has published more than 65 papers in the prestigious journals of the field.

**Dr. Rajani Kumari** is currently an assistant professor at IBS, Bangalore, Off-Campus Centre of ICFAI Foundation for Higher Education (IFHE) University, India. Previously she was an assistant professor at IIIM, Jaipur, and St. Xavier's College Jaipur, CHRIST University. She received the Ph.D. degree in computer science in 2015, the M.C.A. and B.C.A. from University of Rajasthan in 2010 and 2006, respectively. She has published more than forty research papers in various international journals/conferences and participated in many national and international conferences and workshops. She edited some special issue in Taylor & Francis and Inderscience journals including *Journal of Information and Optimization Sciences (JIOS)* and *International Journal of Intelligent Information and Database Systems (IJIIDS)*. Her research interests include Nature-Inspired Algorithms, Swarm Intelligence, Soft Computing, and Computational Intelligence.

## Contributors

**Piyush Aggarwal** Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

**Md Iftekhar Ahmad** National Institute of Technology Patna, Patna, India

**Nasim Akhtar** DUET-Dhaka, University of Engineering & Technology, Gazipur, Bangladesh

**Haider Al-Kanan** Department of Medical Instruments Technology, Al-Kut University College, Alhay, Wasit, Iraq

**Thahab M. AlBuhairi** Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Kingdom of Saudi Arabia

**A. Alfred Kirubaraj** Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India

**Haya A. Alhakbani** Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Kingdom of Saudi Arabia

**Mohd Ali** Department of Computer Sciences, Baba Ghulam Shah Badshah University, Rajouri, India

**Ahmed S. Alzuhairi** Department of Medical Instruments Technology, Al-Kut University College, Alhay, Wasit, Iraq

**J. Andrew** Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India

**K. P. Anu** Indian Institute of Information Technology, Kottayam, India

**Anuneshwar** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**Anudeep Arora** Kamal Institute of Higher Education and Advance Technology, GGSIPU, New Delhi, India

**Anjusha Asok** Christ University, Bengaluru, Karnataka, India

**Sadiya Bashir** Department of Computer Sciences, Baba Ghulam Shah Badshah University, Rajouri, India

**Sanjam Kaur Bedi** Data Science and Artificial Intelligence, International Institute of Information Technology Naya Raipur, Raipur, India

**Momotaz Begum** DUET-Dhaka, University of Engineering & Technology, Gazipur, Bangladesh

**Navneet Bhargava** Malaviya National Institute of Technology, Jaipur, India

**J. V. Bibal Benifa** Indian Institute of Information Technology, Kottayam, India

**Garima Bisht** Department of Mathematics, Statistics and Computer Science, G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India

**Arunkumar Bongale** Department of Robotics and Automation Engineering, Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Pune, Maharashtra, India

**D. Chandrasekhar Rao** Department of IT, Veer Surendra Sai University of Technology, Burla, Odisha, India

**Vidhan Chasta** Department of CSE Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India

**Ankur Chaurasia** Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India

**Gunjan Chugh** Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

**D. Daniel** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**J. Deepa** Department of Information Technology, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

**M. Dhilsath Fathima** Department of Computational Intelligence, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India

**Beata Dratwińska-Kania** Faculty of Finance, University of Economics, Katowice, Poland

**Aleksandra Ferens** Faculty of Finance, University of Economics, Katowice, Poland

**Leonardo Freitas** University of Liverpool, Liverpool, United Kingdom

**Shahid Mohammad Ganie** AI Research Centre, School of Business, Woxsen University, Hyderabad, India

**Lalit Garg** Department of Computer Information Systems, University of Malta, Msida, Malta

**Namrata Gawande** Pimpri Chinchwad College of Engineering, Pune, India

**R. Gokulapriya** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**Gouri Goyal** Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

**Santwana Gudadhe** Pimpri Chinchwad College of Engineering, Pune, India

**Payal Gupta** Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India

**R. Hariharan** Research Scholar, National Institute of Technology, Trichy, India

**R. Harish** Faculty Members at ICFAI Business School, Bangalore, India; An Off-Campus Center of the ICFAI Foundation for Higher Education (IFHE), Hyderabad, India

**Alomgir Hossain** DUET-Dhaka, University of Engineering & Technology, Gazipur, Bangladesh;

IUBAT-International University of Business Agriculture and Technology, Dhaka, Bangladesh

**Nikhil Jain** Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India

**Md Asif Jamal** Department of Computer Science and Information Technology, Maulana Azad National Urdu University, Hyderabad, India

**K. Jayanthi** Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tiruchirappalli, Tamil Nadu, India

**J. Jenefa** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**Jeffin Joseph** Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India

**P. Julian Benadit** CHRIST (Deemed to be University), Bangalore, Karnataka, India

**Harleen Kaur** Data Science and Artificial Intelligence, International Institute of Information Technology Naya Raipur, Raipur, India

**Ranjeeta Kaur** Kamal Institute of Higher Education and Advance Technology, GGSIPU, New Delhi, India

**Manbha Kharsyiemlieh** Department of Information Technology, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

**Manpreet Kaur Khurana** Malaviya National Institute of Technology, Jaipur, India

**Ankush Kumar** Department of CSE Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India

**Pradeep Kumar** Department of Computer Science and Information Technology, Maulana Azad National Urdu University, Hyderabad, India

**Saurabh Kumar** Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan, India

**Subodh Kumar** Department of Electronics and Communication Engineering, Pragati Engineering College, Surampalem, A.P, India

**Sumit Kumar** Arya College of Engineering and IT, Jaipur, India

**Vaibhav Kumar** Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

**Sarita Kumari** Department of Computer Science and Engineering, Banasthali Vidiya Peeth, Rajasthan, India

**Joseph Varghese Kureethara** Christ University, Bengaluru, Karnataka, India

**Manjunath Ramanna Lamani** CHRIST (Deemed to be University), Bangalore, Karnataka, India

**Tripti Lamba** Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

**Eleni G. Makri** Unicaf, Larnaca, Cyprus

**Majid Bashir Malik** Department of Computer Sciences, Baba Ghulam Shah Badshah University, Rajouri, India

**K. Mallikharjuna Rao** Data Science and Artificial Intelligence, International Institute of Information Technology Naya Raipur, Raipur, India

**Sheena Mathew** School of Engineering, Cochin University of Science and Technology, Kerala, India

**Tanvi Mehta** Pimpri Chinchwad College of Engineering, Pune, India

**Rohit Mittal** Manipal University Jaipur, Dahmi Kalan, Rajasthan, India

**Samruddhi Multaikar** Pimpri Chinchwad College of Engineering, Pune, India

**Priyanka Narad** Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India

**R. Nareshkumar** Department of Networking and Communication, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

**Anand Nayyar** School of Computer Science, Faculty of Information Technology, Duy Tan University, Da Nang, Vietnam

**K. Nimala** Department of Networking and Communication, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

**A. K. Pal** Department of Mathematics, Statistics and Computer Science, G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand, India

**Vibhakar Pathak** Arya College of Engineering and IT, Jaipur, India

**Srushti Patil** Pimpri Chinchwad College of Engineering, Pune, India

**Nitin Pise** Dr. Vishwanath Karad, MIT World Peace University, Pune, India

**Vijay Prakash** Department of Computer Information Systems, University of Malta, Msida, Malta;

School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India

**G. N. Raghavendra Rao** Faculty Members at ICFAI Business School, Bangalore, India;

An Off-Campus Center of the ICFAI Foundation for Higher Education (IFHE), Hyderabad, India

**B. T. Ramesh** Department of Robotics and Automation Engineering, Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Pune, Maharashtra, India

**S. R. Jino Ramson** GlobalFoundries US LL2, Vermont, USA

**Aditya Ranade** Dr. Vishwanath Karad, MIT World Peace University, Pune, India

**Utkarsh Rastogi** Department of Robotics and Automation Engineering, Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Pune, Maharashtra, India

**Monika Roopak** Artificial Intelligence and Cyber Security, Department of Computer Science, School of Computing and Engineering, University of Huddersfield, Huddersfield, UK

**Himanshu Bhusan Sahoo** CAPGS, Biju Patnaik University of Technology, Rourkela, Odisha, India

**Ashok Kumar Saini** Department of CSE, SCSE, Manipal University Jaipur, Jaipur, Rajasthan, India

**G. L. Saini** Manipal University Jaipur, Dahmi Kalan, Rajasthan, India

**Madan Lal Saini** Department of CSE Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India

**Rakoth Kandan Sambandam** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**S. Sambhu Prasad** Department of Mechanical Engineering, Nadimpalli Satyanarayana Raju Institute of Technology, Visakhapatnam, A.P., India

**Jibok Sarmah** Department of CSE Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India

**Rahul Saxena** Department of Instrumentation and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka, India

**Javed Sayyad** Department of Robotics and Automation Engineering, Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Pune, Maharashtra, India

**Abhishek Sengupta** Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India

**S. Senith** Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India

**Jankiballabh Sharma** Department of Electronics Engineering, Rajasthan Technical University, Kota, India

**Shweta Sharma** Malaviya National Institute of Technology, Jaipur, India

**K. Sheena Kurian** School of Engineering, Cochin University of Science and Technology, Kerala, India

**Sayar Singh Shekhawat** Department of CSE, SCSE, Manipal University Jaipur, Jaipur, Rajasthan, India

**Sachi Shome** Department of Information Technology, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

**Vishal Srivastava** Arya College of Engineering and IT, Jaipur, India

**Pardeep Singh** Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun, India

**D. Sirisha** Department of Computer Science Engineering, Nadimpalli Satyanarayana Raju Institute of Technology, Visakhapatnam, A.P., India

**Nainsi Soni** Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan, India

**Somaraju Suvvari** National Institute of Technology Patna, Patna, India

**Anuradha Thakare** Pimpri Chinchwad College of Engineering, Pune, India

**Neha Tomer** ICFAI Business School, ICFAI University, Dehradun, Uttarakhand, India

**Van-Truong Truong** Faculty of Electrical-Electronic Engineering, Institute of Research and Development, Duy Tan University, Da Nang, Vietnam

**Amrita Upadhyaya** Department of Computer Science and Engineering, Banasthali Vidiya Peeth, Rajasthan, India

**Michael Vassallo** University of Liverpool, Liverpool, UK

**Prashant Vats** Department of CSE, SCSE, Manipal University Jaipur, Jaipur, Rajasthan, India

**Divya Vetriveeran** Department of CSE, SOET, CHRIST (Deemed to be University), Bangalore, India

**Arpita Vyas** Department of Electronics Engineering, Rajasthan Technical University, Kota, India

# Spam Email Image Detection Using Convolution Neural Network and Convolutional Block Attention Module



Md Asif Jamal<sup>ID</sup> and Pradeep Kumar<sup>ID</sup>

**Abstract** Image spam is continually a popular area of research. Cyberspace is under attack from many different directions. Spam that contains text embedded in an image is known as “image spam”. The rise in online conversation through email has globally contributed to the increasing rate of spam email relatively. First started text spam then now a new challenge in few years image spam which has been a major problem in the field of computing. Various machine learning techniques are used to classify image spam based on a large number of attributes retrieved from the image. Most existing image spam filtering systems use features created by hand and time-consuming machine learning approaches. Convolution neural networks (CNNs) are commonly utilized in image-processing, classification, and feature extraction applications due to their outstanding results. In this research, we use a two CNN model built using deep learning methods to analyze image spam; one is without attention, and next one is with an attention module that time. We use the convolutional block attention module (CBAM); this module attention only spams area of the image and activities of the better performance. Our proposed method achieved very competitive performance—99.42% accuracy on the Image Spam Hunter (ISH) dataset and state-of-the-art performance; in this paper, we used the ISH dataset.

**Keywords** Image spam · Deep learning · Convolution neural network (CNN) · Convolutional block attention module (CBAM)

---

M. A. Jamal (✉) · P. Kumar

Department of Computer Science and Information Technology, Maulana Azad National Urdu University, Hyderabad, India  
e-mail: [asifjamal29@gmail.com](mailto:asifjamal29@gmail.com)

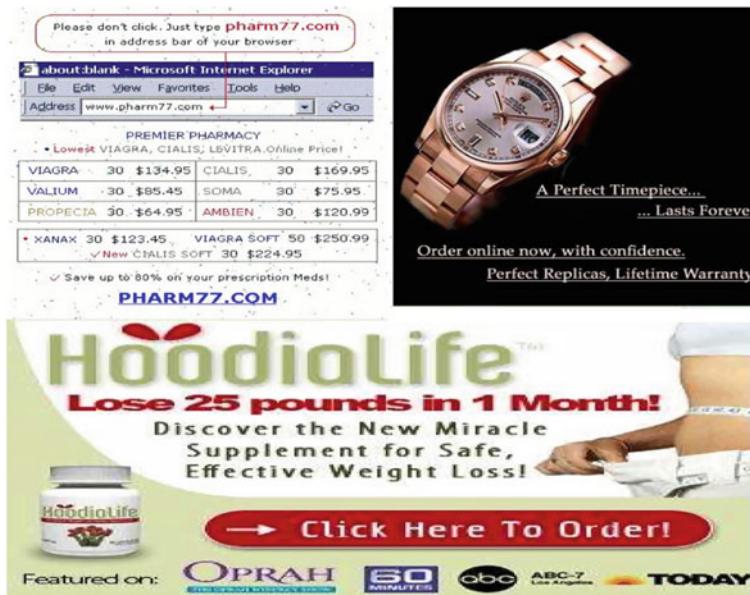
P. Kumar  
e-mail: [pradeep@manuu.edu.in](mailto:pradeep@manuu.edu.in)

## 1 Introduction

One of the biggest issues on the Internet is spam. Over the past few years, spam has greatly expanded in volume. More than 80% of emails and messages people get nowadays are spam [1].

Today smart phone is common for people which is a big reason to increase an email user. Through email, we connect all over the word and communicate business and individual purposes. Spam mail means undesirable email comes in our inbox like some advertisement of some products and bulk email like weight loss advertisement, some product discount advertisement, etc. When text is included in an image, it is known as image spam for the purpose of avoiding text-based spam email screening systems. Because of the sort of information they contain, image detection may be a difficult task for both machines and people. An effective solution for the detection of image spam is therefore required.

Initially, machine learning (ML)-based algorithms' spam detectors were developed for filtering text-based email spam at an average 90% accuracy [2] using support vector machine (SVM), K-Nearest Neighbors (KNNs), and Naïve Bayes (NB) models. As spamming strategies advance, ongoing study and development of advanced spam detection algorithms are necessary. As spammers create new methods to circumvent based on text spam filters, they began using image spam to deliver unwanted messages. This type of spam involves delivering textual information in the form of images. This is shown in Fig. 1.



**Fig. 1** Sample of spam image

Deep learning (DL)-based methods take unprocessed imagery as input since they have the ability to automatically extract features from the raw images. CNN is one of the deep learning methods. When applied for image classification, the network stands out, resulting in various improvements to deep network training. A DL model needs a lot of data to be trained because it has millions of trainable parameters, and a small dataset wouldn't be enough to achieve good generalization. Therefore, we suggest using a pretrained CNN model that applies the transfer learning (TL) approach as a spam image feature extractor. The collected features are then passed into our Re-Dense layer that has been adjusted and optimized, which determines if the input image is spam or not.

Related work to the image detection of a spam is included in the section that follows relating to the work mentioned in Sects. 2 and 3 which are CNN modules. Information about the CBAM is presented in Sect. 4. We present the experimental findings in Sect. 5. In this paper conclusion, Sect. 6 makes suggestions for potential areas of future study.

## 2 Related Works

The classification of input into spam or non-spam can be done in two ways for picture spam detection. The first method is based on extracting the text that is included into the picture applying optical character recognition (OCR) algorithms. The second method for classifying picture spam makes use of a variety of visual attributes and features that are based on both ML and DL techniques.

DL has many fields, including natural language processing (NLP), pattern recognition, and computer vision, and these techniques have produced positive outcomes. In order to classify visual spam, one of the most effective deep learning techniques is the CNN. The CNN model's design primarily includes three layers. Multiple convolutional and pooling layers are repeated multiple times between the input and output layers.

According to [3], a problem for online communication platforms is image spam, which is a sophisticated type of unwanted email created to get past text-based spam filters. The hand-crafted features and ML used in current picture spam filtering algorithms are time-consuming, less effective, and frequently missing important features. With the use of the Image Spam Hunter dataset, this paper proposes the CBAM model for image spam identification. Even though the results are better than those of earlier methods, future research should explore other deep learning models and add more hyperparameters, such as combining edge detector methods like Canny edge with recurrent neural networks (RNNs) and long short-term memory (LSTM).

To obtain past text-based spam filters, spammers often embed spam content into graphical graphics, as discussed in this essay. The methods currently in use concentrate on the image's textual or graphic elements but are ineffective in identifying image spam. The authors suggested a framework named SPAMI to fill this gap by labeling spam advertisement images using DL models CNN, RNN, and LSTM. With

real-time gathered photos, the system extracts feature with an accuracy of 95%, while with the “Image Spam Hunter” dataset, it achieves 97% accuracy. Future studies will focus on extracting more practical features and investigating new deep learning models for better spam advertisement image detection [4].

This research focuses on identifying spam emails, which cause enormous financial losses. For spam identification, a variety of ML and DL algorithms have been used, with NLP approaches increasing accuracy. In order to categorize spam emails, the model bidirectional encoder representation from transformers (BERT) is used in the study to demonstrate the efficiency of word embedding. The findings, which achieve the greatest accuracy of 98.67% and an F1 score of 98.66% when compared to baseline models such as deep neural network (DNN), KNN, and NB classifiers, are presented. Because it takes context into account through attention layers, the BERT-base-cased transformer model performs better than other methods. Larger input sequences can be investigated in the future, and spam identification can be expanded to include languages like Arabic [5].

A hybrid deep convolution neural network (DCNN) made up of the convolution neural networks CNN1 and CNN2 was proposed by Anivila and Soman in 2020 [6]. The DCNN was trained with three datasets in the study, and transfer learning was also investigated by utilizing established frameworks like VGGC19 and ImageNet throughout the training stage. They displayed a result with an accuracy of 97.1% and an F1-score of 97.4%. This study addresses the issue of image email spam filtering, which is challenging because spammers use obfuscation techniques. The suggested method, known as Image Texture Analysis-Based Spam Filtering (ITA-ISF), uses low-level image properties to characterize images. With an average precision, recall, accuracy, and F-measure of 98.6%, the Random Forest (RF) classifier outperforms other ML classifiers. Utilizing feature selection to condense the feature space, the technique retrieves the features linked to the image texture. In addition, the SVM classifier’s kernel parameter can be altered to reduce the number of false positives [7].

In this paper, a three-layer image spam filtering method was introduced. To identify spam, email header and attachment analysis are performed. The primer layer, which focuses on the message header, successfully recognizes 93.7% of the image mails, greatly reducing the processing time. The second and third layers examine high-level and low-level image features, respectively. According to the experimental findings, the system had a 94% accuracy rate, indicating its efficiency and usefulness in removing picture spam. Overall, the multi-layer method described offers a potential way to stop a picture spam [8].

Also, Nikhil Kumar, Sanket Sonowal, and Nishant [9] discuss the growing issue of email spam, which is used for fraud, phishing, and unlawful activity. The objective is to use machine learning to detect phoney spam emails. Datasets are subjected to a variety of methods, and the optimal approach is chosen based on the precision and accuracy. While performing well, the multinomial naive Bayes method has limitations in its ability to correctly identify some emails. Through the employment of several classifiers for prediction, ensemble approaches are beneficial. Instead of focusing on domain names, the project focuses on filtering emails based on content. Filtering based on reputable domain names and classifying emails as spam or not

is a possible future enhancement. Organizations may find this strategy useful for removing spam emails.

This paper [10] focuses on the identification of picture spams, a danger to email-based communication, in emails. By training on different image features, SVM is utilized to analyze image spam detection. The usefulness of machine learning algorithms based on image-processing techniques was demonstrated using real-world datasets. Three datasets are used in this paper.

1. A sizable sample of spam photographs and an equally sizable sample of ham images were gathered using Image Spam Hunter Gao et al. [16]. The ISH dataset is the collection of this data. A total of 920 spam photos and 810 ham images from the ISH dataset were retained after data cleaning. These pictures are all in jpg format.
2. A picture spam corpus was produced by Dredze et al. and is available to the general public. This set is referred to as the Dredze dataset. We retained 1089 spam and 1029 ham photos from the Dredze dataset after cleaning. All photos are in the jpg format, similar to the ISH dataset.
3. However, image spam can undoubtedly be made much harder to spot. Consequently, we created our own challenge dataset. The aim of this dataset is to test the detection of more sophisticated types of picture spams, which are likely to appear in the near future. To make spam photographs appear more like ham images, we used a variety of image editing techniques on real spam images. For our spam corpus, we used the Dredze dataset and then overlaid ham pictures from the ISH dataset. This study identifies the shortcomings of existing image spam detection techniques and suggests the need for new strategies. Researchers are given access to a challenge dataset for additional analysis and comparison with other detection methods, such as deep learning and neural networks.

Annadatha and Stamp [11] utilized a variety of extracted picture attributes, including color, edges, local binary patterns (LBPs), histograms of oriented gradients (HOGs), and ratio of compression. To detect picture spam, they used a linear SVM and eigenspam. On one dataset, the experimental findings performed well; however, on the upgraded dataset, they performed only 38% better. In addition, [12] suggested combining hue saturation value (HSV) and RGB histograms, two distinct types of characteristics. The KNN technique uses these qualities to identify the image span, and the accuracy of the proposed method was 94.5%.

The problem of removing spam photos from emails was addressed in this study. To differentiate spam images from text from other types of images, a hybrid strategy that incorporates features from text areas and image features is suggested. For the picture texture and text regions, features were retrieved using the local binary pattern (LBP). To train one- and two-class KNN classifiers, the retrieved features are employed both alone and in combination. According to the experimental findings, integrating picture and text characteristics enhances classification accuracy compared to using only one or the other. The text identification technique used was effective even for low-contrast photographs and a variety of font sizes and styles, providing clear cues for identifying spam from legitimate images [13].

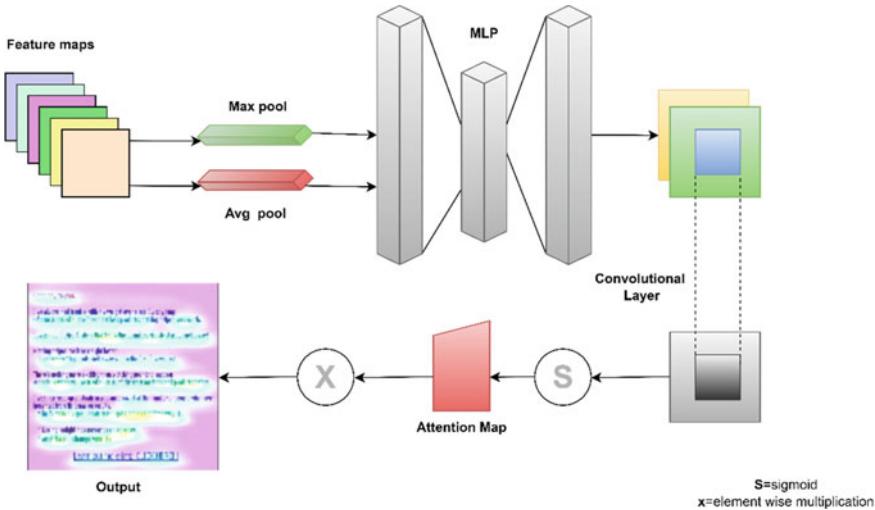
The classification of picture spams using DL and CNN is the main goal of this study. On several image spam datasets, the proposed CNN model outperformed competing methods for feature extraction and classification tasks. The findings demonstrate that excellent classification accuracy can be attained using deep learning approaches, particularly when combined with transfer learning, to efficiently extract features from raw input photos. The improved model exhibits potential for use in additional picture categorization domains. Future research should investigate automatic parameter-tuning techniques and assess how well the suggested algorithm performs on other image datasets for statistical analysis [14].

### 3 Convolution Neural Network

Two DCNN models are generated in this work. CNN1's initial model consists of three convolution layers with filter widths of 32, 64, and 128. The second CNN module is connected to the CBAM (attention module). It has also three filter sizes: 32, 64, and 128. A straightforward but efficient feed-forward CNN attention module. Our module is sequentially analyzed by many techniques. The efficacy of the CNN model in tasks such as classification has been enhanced using the methods; we apply the attention mechanism. The human visual system-based attention mechanism is employed to boost CNN's power for representation. Humans are distinguished from conventional neural network models by their capacity to concentrate on key details rather than analyze the full scene. The CNN model includes the attention module to concentrate on important elements and exclude irrelevant ones. To combat this problem, we implement the CBAM in this study. Module can be end-to-end trained alongside base CNNs and is easily implemented into any CNN architecture with minimal overhead.

Figure 2 shows how much emphasis the initial CNN feature map placed on. Based on our findings, it appears that our model can highlight relevant features, hide irrelevant ones, and extract relevant data from the feature map. The following are some of the major contributions of this work.

- (1) We propose a straightforward and powerful attention module that can be widely deployed to improve CNN's representational prowess.
- (2) Our strategy makes scene classification of spam photos containing numerous tiny objects and complicated backdrops easier.
- (3) By embedding the attention module, we achieve state-of-the-art performance on Image Spam Hunter (ISH) classification datasets.



**Fig. 2** Convolutional neural network with convolutional block attention module

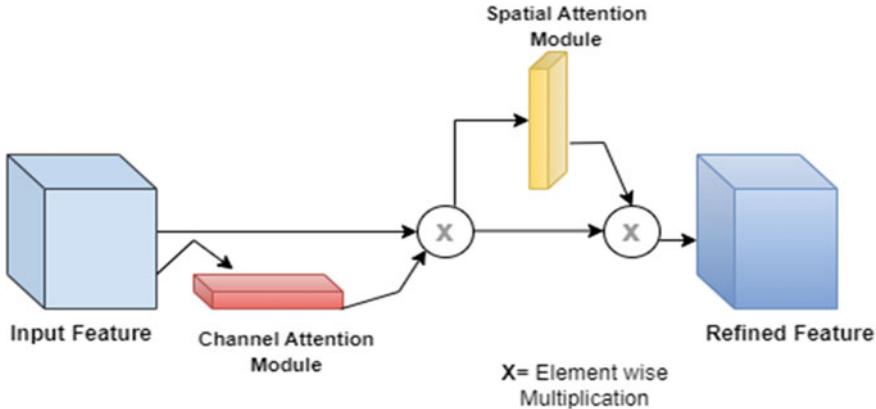
## 4 Convolutional Block Attention Model

The vast representational power of CNN enhances the performance of numerous visual tasks. Three factors depth, width, and cardinality have mostly been examined by researchers to improve the quality of CNNs. Aside from these three things, the attention mechanism has been thoroughly studied in various studies. This feature of architectural design is used to the representation strength is increased by highlighting key elements and omitting minor ones. Using the CBAM [15].

To enhance the CNN model's attention mechanism, the paper proposes using the CBAM module that integrates both the channel attention module (CAM) and the spatial attention module (SAM). CAM focuses on interdependencies among feature channels, while SAM emphasizes interdependencies among feature map locations. By combining CAM and SAM, the CBAM module can obtain more comprehensive attention information. As a result, the model can allocate computing resources more effectively, leading to improved classification performance without a significant increase in computation. Therefore, the CBAM mechanism is a promising solution to enhancing CNN models' accuracy.

In Fig. 3, CBAM module applies a two-stage process to refine the feature map  $F$ . In the first stage, to create the channel attention map  $M_c$ ,  $F$  is passed through the channel attention module. This map captures the interdependencies among the different feature channels.

In the second stage, the channel attention feature map  $F'$ , which is the element-wise product of  $F$  and  $M_c$ , is used to create the spatial attention map by the spatial attention module,  $M_s$ . This map captures the interdependencies among the different spatial locations in the feature maps.



**Fig. 3** Convolutional block attention model

The CBAM algorithm sequentially calculates a 1D channel attention map  $M_c \in \mathbb{R}^{C \times 1 \times 1}$  and a 2D spatial attention map  $M_s \in \mathbb{R}^{1 \times H \times W}$  for a given feature map  $F \in \mathbb{R}^{C \times H \times W}$ . These attention maps are then used to improve the output  $F''$ . Based on the findings presented in the referenced study, it is observed that the sequential approach often yields superior outcomes when compared to the parallel approach. The procedure for integrating the two attention modules into any convolutional neural network (CNN) model is outlined as follows.

$$F' = M_C(F) \otimes F \quad (1)$$

$$F'' = M_S(F') \otimes F' \quad (2)$$

This symbol  $\otimes$  is indicated to the element-wise multiplication,  $F^c_{\text{avg}}$  is known as average-pooled features, and  $F^C_{\text{max}}$  are denoted as max-pooled features.

The utilization of the channel attention module is employed when presented with an input image, to direct attention toward significant aspects and capitalize on the interconnectedness between channels within the features. Within this module, the process of generating the average-pooled features  $F^c_{\text{avg}}$  and the max-pooled features  $F^C_{\text{max}}$  from the input feature map involves the consecutive application of global average pooling and global max pooling operations. The input of a shared multi-layer perceptron (MLP), consisting of one hidden layer, comprises these two descriptors. The allocation of attention to the different channels is provided in the following manner.

$$\begin{aligned} M_C(F) &= \sigma(\text{MLP}(\text{avgpool}(F)) + \text{MLP}(\text{Maxpool}(F))) \\ &= \sigma(w_1(W_0(F_{\text{cavg}})) + w_1(w_0(F_{\text{cmax}}))) \end{aligned} \quad (3)$$

**Table 1** Dataset

S. No.	Category of image	Number of images
1	Spam images	928
2	Ham images	810
3	Total numbers of images in dataset	1738

The weights of the shared MLP are denoted as  $W_0$  and  $W_1$ , and they utilize the rectified linear unit (ReLU) activation function. The symbol  $\sigma$  indicates the sigmoid function. In contrast, the spatial attention channel leverages the inter-spatial relationship of the features. In contrast to directing attention, this module focuses on the location or presence of information components. Two distinct features are produced from the channel information, specifically the average-pooled  $F^s_{\text{avg}} \in \mathbb{R}^{1 \times H \times W}$  and the max-pooled  $F^s_{\text{max}} \in \mathbb{R}^{1 \times H \times W}$ . To acquire a 2D spatial attention map, the aforementioned features were combined and subjected to convolution using a typical convolution layer.

$$\begin{aligned} M_s(F) &= \sigma(f7 \times 7([ \text{AvgPool}(F); \text{MaxPool}(F) ])) \\ &= \sigma(f7 \times 7([ F_{\text{avg}}; F_{\text{max}} ])) \end{aligned} \quad (4)$$

$\sigma$  is indicated as sigmoid function or  $f7 \times 7$  indicated as size of  $7 \times 7$  convolution operation.

## 5 Experimental Result

### 5.1 Dataset

We use those datasets which have publicly available. This dataset in Table 1 is referred as ISH. A group of North western university [16] created these ISH datasets. All images are in the jpg formats.

### 5.2 Evaluation Metrics

Using datasets, the CBAM concept is contrasted with current methods for identifying spam photos. Accuracy, precision, recall, and F1-score are the four evaluation metrics we use to assess our suggested technique. Following is how these metrics are calculated.

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (5)$$

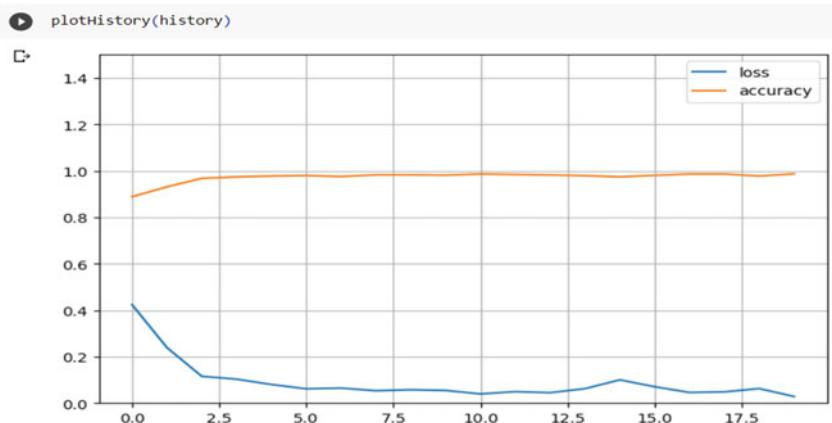
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (6)$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (7)$$

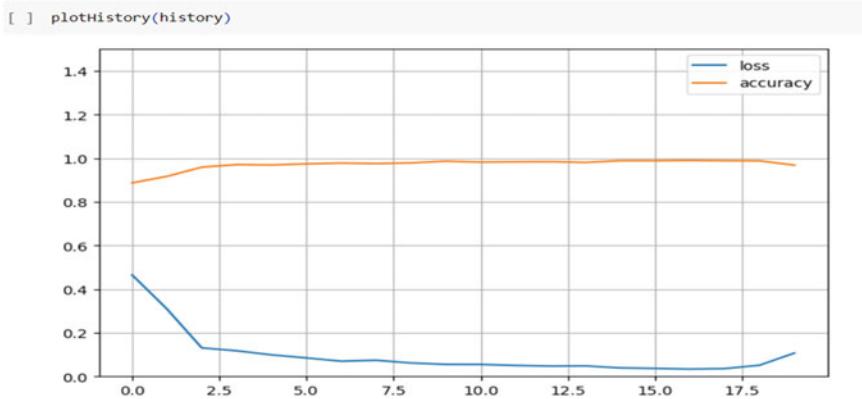
$$\text{F1Score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (8)$$

In the domain of spam image detection, the terms True Positives (TP), False Negatives (FN), False Positives (FP), and True Negatives (TN) correspond to specific quantities. TP refers to the count of spam images that are accurately classified as such. FN represents the number of spam images that are inaccurately classified as non-spam. FP denotes the count of non-spam images that were mistakenly classified as spam. Finally, TN signifies the count of non-spam images that are correctly classified. Figures 4 and 5 show loss and accuracy graph.

To begin our proposed method for detecting image spam, the first step involves normalizing and resizing the images. This is necessary because the datasets used in our experiments include images of varying sizes. Therefore, to ensure consistency and ease of processing, we resize all images to a uniform size of  $128 \times 128$ . In this research, we use a CNN model with six convolution layers and the CBAM. A total of 32 filters are used in the first and second convolution layers of the CNN architecture. The following step includes two convolutional layers with 64 filters each. 128 are present in the five and six convolutional layers. We use an Elu activation function and a kernel of size  $3 \times 3$  for all convolutional layers. The max pooling layer, which has a  $2 \times 2$  pool size, is used before the second and fourth convolutional layers.



**Fig. 4** With attention number of CNN loss of model and accuracy of model of Image Spam Hunter dataset



**Fig. 5** Without attention number of CNN loss of model and accuracy of model of Image Spam Hunter dataset

**Table 2** Comparison of accuracy among others models

Heading level	Accuracy (%)
CNN with CBAM	<b>99.42</b>
CNN without	85.59
Chavda et al. (SVM) [10]	97.00
Ghizlane et al. with attention[3]	98.65
Srinivasan et al. [6]	97.10
Al-Duwairi et al. (texture analysis) [7]	98.60
Liu and Tsao et al. [8]	94.00

Even in the final convolutional layer we use a global average pooling layer, the batch normalization is still applied after each convolutional layer. The CNN model successively infers the CBAM. The results obtained in this paper using the CBAM model are compared to the existing methods, as shown in Table 2.

### 5.3 Discussion

In this work, the dataset is divided in two, with 80% of the training set and 20% of the testing set, respectively. The Sigmoid and Adam functions are used to train the CBAM model over the course of 20 epochs. The results achieved using the measures mentioned above are then contrasted with the most recent deep learning models for email spam detection. The findings of the CBAM model's analysis of the ISH dataset are presented in Table 1, along with comparisons to current state-of-the-art techniques. Table 1 shows that the ISH dataset accuracy of the CNN with CBAM

model was 99.42% shown in Fig. 4, while the accuracy of the CNN without CBAM model was 85.59% shown in Fig. 5.

## 6 Conclusion

Because of its nature, spam picture identification is regarded as a challenging subject. The ISH dataset was used in this research to test the CBAM. The reported results, which somewhat outperform those from earlier study, demonstrated the efficacy of the suggested strategy. For the detection of spam images, the CBAM yielded good results; however, different deep learning models and extra hyperparameters can improve these findings. To examine the context of spam images, in future we will use a huge dataset of spam images, a transfer learning model, and other deep learning techniques including RNN and long short-term memory (LSTM).

## References

1. Abuzaid N (2022) Image spam detection using ML and DL techniques. *Int J Adv Soft Comput Appl* 14(1):227–243
2. Lai CC, Tsai MC (2004) An empirical performance comparison of machine learning methods for spam e-mail categorization. In: Fourth international conference on hybrid intelligent systems (HIS'04). IEEE, pp 44–48
3. Ghizlane et al (2022) Spam image detection based on convolutional block attention module. In: International conference on intelligent systems and computer vision (ISCV)
4. Makkar A, Kumar N, Zomaya AY, Dhiman S (2020) SPAMI: a cognitive spam protector for advertisement malicious images. *Inf Sci* 540:17–37
5. AbdulNabi I, Yaseen Q (2021) Spam email detection using deep learning techniques. *Procedia Comput Sci* 184:853–858
6. Srinivasan S, Vinayakumar R, Vishvanathan S, Krichen M, Noureddine D, Anivilla S, Soman KP (2020) Deep convolutional neural network-based image spam classification. In: 6th conference on data science and machine learning applications (CDMA). Riyadh, Saudi Arabia, pp 112–117
7. Al-Duwairi, Khater I, Al-Jarrah O (2012) Detecting image spam using image texture features. *Int J Inf Secur Res (IJISR)* 2(3/4):344–353
8. Liu TJ, Tsao WL, Lee CL (2010) A high performance image-spam filtering system. In: Proceedings of the 2010 ninth international symposium on distributed computing and applications to business engineering and science. IEEE, Hong Kong, China, pp 445–449
9. Kumar N, Sonowal S, Nishant (2020) Email spam detection using machine learning algorithms. In: 2020 second international conference on inventive research in computing applications (ICIRCA)
10. Chavda A, Potika K, Troia FD, Stamp M (2018) Support vector machines for image spam analysis. In: Proceedings of the 15th international joint conference on e-business and telecommunications, vol 1. BASS, Porto, Portugal, pp 431–441
11. Annapurna A, Mark S (2016) Image spam analysis and detection. *Artic J Comput Virol Hacking Tech*
12. Kumaresan T, Sanjushree S, Palanisamy C (2015) Image spam detection using color features and K nearest neighbor classification

13. Dhah EH, Naser MA, Ali SA (2019) Spam email image classification based on text and image features first. Int Conf Comput Appl Sci (CAS), 978-1-7281-4048-3/19/\$31.00 ©2019 IEEE
14. Singh AB, Singh KM, Chanu YJ, Thongam K, Singh KJ (2022) An improved image spam classification model based on deep learning techniques. Hindawi Secur Commun Netw 2022, Article ID 8905424
15. Woo S, Park J, Lee JY, Kweon IS (2018) CBAM: convolutional block attention module. Lect Notes Comput Sci (including Subser. Lect Notes Artif Intell Lect Notes Bioinformatics) 11211:3–19. LNCS
16. Gao Y, Yang M, Zhao X (2008) Image spam hunter EECS Dept., North western Univ. 2145 Sheridan Rd., Evanston, IL 60208 {y-gao2, m-yang4, [xiaonan-zhao@northwestern.edu](mailto:xiaonan-zhao@northwestern.edu)}

# Identifying Fake Twitter Trends with Deep Learning



Thahab M. AlBuhairi and Haya A. Alhakbani

**Abstract** This paper describes a model for detecting and distinguishing fake tweets from real tweets. The model uses one of the most commonly used deep learning algorithms in natural language processing, the Bidirectional Long Short-Term Memory (Bi-LSTM) algorithm. The model was trained on two datasets, the PolitiFact dataset from the FakeNewsNet repository and the PHEME dataset. The datasets are divided into 80% for training and 20% for testing. The model contains two merged models (hybrid model), one for text attributes and the other model for numeric/categorical attributes. To achieve the best result, the model balanced the datasets using four balancing techniques: RandomOverSampler, SMOTE, RandomUnderSampler, and NearMiss. In addition, there are two combination strategies that combine two strategies with certain percentages: combineOverUnderSampling and combineSMOTEUnderSampling balancing strategies. This work is one of the first works to use these combinations and obtain very good results, more than most related studies. For the PHEME dataset, the model was trained with a variety of epochs (10, 20, 50, and 100) to obtain the best result. For the PolitiFact dataset, only 10 epochs were trained at a time. The best result obtained with the PolitiFact dataset is 0.96 for accuracy and precision, 0.95 for recall, F1-score, and ROC-AUC. The best result obtained with the PHEME dataset is 0.87 for accuracy and 0.86 for precision and 0.85 for recall, F1-score, and ROC-AUC.

**Keywords** Fake · Tweets · Twitter · Trends · Deep learning · NLP · AI · Bi-LSTM

---

T. M. AlBuhairi (✉) · H. A. Alhakbani

Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Kingdom of Saudi Arabia  
e-mail: [amalbuhairi@imamu.edu.sa](mailto:amalbuhairi@imamu.edu.sa)

H. A. Alhakbani  
e-mail: [hahakbani@imamu.edu.sa](mailto:hahakbani@imamu.edu.sa)

## 1 Introduction

Recently, most people used smartphones every day and can browse social media anytime and anywhere. Therefore, the number of social media users is increasing tremendously and rapidly. Twitter (currently X) is considered the 12th most popular social media platform in the world as published in 2019, with more than 330 million active users [1] and growing to 436 million users in 2022 [2] and more than 500 million tweets per day [3]. This shows how these people influence society and express their opinions. Twitter users can access news and personal opinions about any trending event. However, some users have ulterior motives to manipulate Twitter tweets and news that influence the society. These groups may use some hacking methods to spread fake news and tweets in an illegal way, without enough legitimate tweets or with non-relevant content. For this reason, this paper aims to build a model using deep learning to distinguish fake tweets from real tweets and eliminate this problem. Deep learning is a subset of machine learning architecture in which structures are built using artificial neural models. Deep learning structures such as recurrent neural networks are used in a variety of domains, including natural language processing (NLP), to produce results that are superior to human capabilities.

The contributions of this paper are the following:

- Utilizing the combination of two balancing techniques. As far as we know, this paper is the sole attempt to achieve this.
- Comparing the recent model with the previous works on two datasets—PolitiFact from FakeNewsNet repository and PHEME 8 events dataset.
- The proposed model outperforms most of the current state-of-the-art models for both datasets.

The rest of the paper is organized as follows. Section 2 examines some related work. Present the methodology and used model in Sect. 3. Discuss and analyze the results in Sect. 4. Finally, conclude the paper with future work in Sect. 5.

## 2 Related Work

This section focuses on some related works of both the two used datasets.

### 2.1 *PolitiFact Dataset*

The FakeNewsNet repository is new and was presented first in 2019, and little research has been done on it. A study titled “SpotFake+ : A Multimodal Framework for Fake News Detection via Transfer Learning” used this repository and was published in 2020 [4]. SpotFake+ used transfer learning to capture semantic and

contextual information. The SpotFake+ algorithm achieved the best results compared to many other algorithms such as SVM, logistic regression, Naive Bayes, CNN, social article fusion (SAF), XLNet + dense layer + XLNet + LSTM, SpotFake, and others. The accuracy reached 84 and 85% for SpotFake+ algorithm, while the highest accuracy for the others did not exceed more than 74 and 83% [4].

Another study that used the FakeNewsNet repository is [5], which focused on benchmark improvement. N-gram models were created in two ways: base models and hybrid models. The dataset is then preprocessed, and the features are weighted with TF-IDF and analyzed with logistic regression. Prior to analysis, the researchers applied the SMOTE filter to the training set. The n-gram model achieved 80% accuracy, 79% precision, 78% recall, and 79% F1-score when applied to PolitiFact [5].

A study published in December 2021 addressed the detection of fake news in English. The FakeNewsNet dataset was used for this purpose [6]. A binary classification of “fake” or “real” news is performed. PolitiFact contains 624 real news items and 432 fake news items, 80% of which are used for training, 10% for validation, and 10% for testing. This method consists of six parts: text preprocessing, tokenization, and using a Statistical Feature Fusion Network (SFFN) with MCDropout. The “NewBERT” model obtains the prediction vectors using a heuristic algorithm. To solve the imbalance in the dataset, K-Means-SMOTE algorithm was used [7].

The SFFN model with MCDropout achieved the highest performance compared to the other classifiers. It achieved about 91% in accuracy, precision, recall, and F1-score [6].

## 2.2 PHEME Dataset

Kochkina and her colleagues used the PHEME dataset and applied many experiments and approaches, such as sequential and multi-task learning approaches [8]. The experimental setup they used consists of hyperparameters. They used the Tree of Parzen Estimators (TPEs) algorithm and a number of LSTM layers. They performed 30 trials with different parameter combinations to optimize the accuracy of the development set and select the best combination. The Ottawa shooting achieved the best accuracy of 64% [8].

In 2021, researchers in [9] built a hybrid model with a Convolution Neural Network (CNN) and a filter wrapper optimized by the Naive Bayes classifier to detect rumors and use the PHEME dataset focusing on textual features. The proposed deep neural model is CNN-IG-ACONB. Feature selection techniques can be broadly divided into filters, wrappers, and embedded methods. Wrapper techniques assess all possible feature combinations to generate an optimal feature set. The rumor detection process consists of four phases: rumor detection, rumor tracking, stance classification, and

veracity classification. The text is represented as an array of vectors with 1D convolutions in sequential data. In addition, the ELMo 5.5B word vector model and TF-IDF were used for feature selection. The results for the whole dataset are 0.776 for precision, 0.745 for recall, and 0.732 for F1-score.

On the other hand, Shakshi and Rajesh presented a work in the same area to detect the rumor spreaders by examining the sentiments of the tweets and computing the rumor tweets of the user posts [10]. It also uses three sets of features: user features, text features embedded in 300 dimensions of word2vec, and geo-network features. For data cleaning: removing non-alphanumeric characters, URLs, stop words, and punctuation, converting words to lowercase, and using Porter Stemmer for stemming. TextBlob API was used to analyze the sentiments of replied tweets.

They are investigating Graph Convolutional Networks (GCNs). The method used achieved an F1-score of 0.864 and a value of 0.720 in AUC-ROC [10].

This paper works on another model for detecting fake tweets using deep learning methodology and tried in both datasets to achieve a higher score than the related works. In the next section, the methodology used in this work is explained.

### 3 Methodology

In this study, Python is used as the programming language because it is a high-level, open-source, and general-purpose programming language. It is also easy to learn and has many packages and libraries that support the concept of deep learning. The navigation platform Anaconda, Jupyter Notebook version 6.0.1, and Google Colab are used as coding platforms.

#### 3.1 Data Collection

**PolitiFact dataset.** The first version of the dataset comes from an open data repository called FakeNewsNet, generated in 2018; the second version has been available since 2019 on this reference [11]. The creators of this dataset are Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. The dataset was collected and analyzed using the FakeNewsTracker tool [12]. This dataset focuses on political news in news content. It was gathered from a nonprofit website operated by the Tampa Bay Times and then transferred to the Poynter Institute for Media Studies in 2018 [13]. The total number of tweets from this dataset is 431,496 before preprocessing, and this dataset represents the big dataset.

**PHEME dataset.** This dataset was released in 2018; it contains more than 105 thousand Twitter tweets and their reactions related to nine famous events, such as the Ferguson unrest, the Charlie Hebdo shooting, and others between 2014 and 2015 [8, 14]. In this work, only the tweets without the reactions are used; the total number

of tweets from this dataset is 6424 before preprocessing. This dataset represents the small dataset in this work.

The PolitiFact and PHEME datasets are available online for free as a separate JSON file for every single tweet. For the PHEME dataset, this work has focused on the original tweets without the reaction tweets.

The selected attributes are tweet\_id, tweet\_text, source, date of post the tweet, user\_id, user\_name, user\_description, date of create the account, user\_location, followers\_count, friends\_count, favorite\_count, and language. Then, add two additional labels, one named “event” for the name of the event dataset in the PHEME dataset and the second named “target”, filled with 0 for real tweets and 1 for fake tweets. Finally, paste all these information into one CSV file.

The next stage is preprocessing and cleaning the data, which will be explained in more detail in the next section.

### 3.2 Data Preprocessing

- Data cleaning: Remove the incorrect and missing values by removing incomplete instances and replacing abnormal data with particular text.
- Remove all non-English tweets, as this work focuses only on English language.
- Below are two sub-steps of preprocessing the data, the first focusing on the textual fields (text attribute and user description). The second sub-step focuses on the numeric and categorical attributes (source, username, user\_location, followers, followings, retweet\_count, and fav\_count).
- For the textual attributes:
  - Tokenizing the dataset by dividing it into separated words using the TweetTokenizer from the NLTK library.
  - Removing URLs, punctuation, and any non-alphabetic characters.
  - Converting all texts to lowercase and replacing inconsistencies.
  - Removing stop words.
  - Lemmatizing words: The used lemmatizer is WordNetLemmatizer.
  - Stemming: The famous current stemmer is the snowball.
  - For numeric and categorical attributes:
  - Using the LabelEncoder to encode the three categorical attributes (source, user\_name, user\_location).
  - The previous three categorical attributes are added to the four numeric attributes (followers, followings, retweet\_count, and fav\_count) to form a numeric vector of 7 lengths.
  - Normalize these data to a value between [0,1] using MinMaxScaler.
- Generate a padding vector of 40 lengths for both the user description and the word indexes of the tweets.

- Create the embedding matrix using the pre-trained GloVe model with version 840B and 300d dimensions. It contains more than 2,196,016 English word vectors [15].
- The number of vectors in PolitiFact dataset is 252439 vectors and 10,750 vectors for PHEME dataset.
- Each dataset divides into 80% for training and 20% for testing.

### 3.3 *Modeling*

The used algorithm is a Bidirectional Long Short-Term Memory (Bi-LSTM). Bi-LSTM deals with supervised classification and takes care of the input sequence. It trains two models instead of one LSTM on the input sequence, one in the backward layer and the other in the forward layer, which can remember the past and future steps [16]. Since the datasets are not balanced, the model must use a balancing technique to facilitate the work. In this work, six balancing techniques were used, which are explained in more detail below.

**Balancing the dataset (oversampling and undersampling).** The data in any given dataset must be nearly balanced between classes in order to give the machine fair learning opportunities when learning and to prevent bias against any class. This is done by increasing the size of the minor class to be equal to the major class (oversampling) or decreasing the size of the major class to be equal to the minor class (undersampling).

Four main strategies have been tried for balancing in this work, two for oversampling: RandomOverSampler and Synthetic Minority Oversampling Technique (SMOTE). Two more strategies are for undersampling, RandomUnderSampler and NearMiss.

After that, generating additional two strategies of balancing, which are a combination of two types of balancing techniques: one for oversampling and the other for undersampling:

- Combining over and undersampling: This private procedure adds samples to the minority while deleting samples from the majority according to certain percentages (0.7 for the oversampling part and 0.8 for the undersampling part).
- Combining SMOTE and undersampling: This strategy uses SMOTE for the oversampling and random undersampler for the undersampling strategy, with the same procedure as the previous one.

**Building the model.** In the modeling, the attributes include two different types of data: textual and categorical/numerical. A different learning model was built for each type and the displayed results were then merged as explained below:

#### A. Textual Neural Network Layers

The network aims to build a learning model that classifies textual attributes (tweets text and user description) as fake or real using deep learning techniques (Bi-LSTM). The unique feature of this algorithm is that it works in two directions, forward and backward. This network uses nine layers: Input layer, Embedding layer, Spatial-Dropout1D layer, Conv1D layer, bidirectional LSTM layer, First dense layer, Dropout, Second dense layer, and Third dense layer which leads to the output layer.

## B. Numbers Neural Network Layers

The goal of this network is to build a learning model that has seven tweet description attributes as input after preprocessing them into numbers (source, username, user\_location, followers, followings, retweet\_count, fav\_count) and classify the attribute “target” as fake or not as output by using Deep Learning algorithm (Bi-LSTM) as well.

This network consists of four layers: Bidirectional LSTM layer, First dense layer, Second dense layer and Third dense layer.

The model trained 10 epochs for the PolitiFact dataset and also trained 10, 20, 50, and 100 epochs for the PHEME dataset. An epoch means the time it takes for the data to completely run through an algorithm. The batch size is 1024 and the class “ReduceLROnPlateau” is used for the callbacks, which is useful to reduce the learning rate when a metric stops improving with factor = 0.1.

**Combining learning models (hybrid model).** Combine the results from the previous two networks together. Use a weighted average between each model scores to obtain the final classification score. The weights can vary between textual and numerical attributes according to achieving the goal of this work and obtaining the best results. The weighting is 0.7 for the textual model result and 0.3 for the other model results.

**Performance measures.** The following performance measures are used in this work Accuracy, precision, recall, F1-score, and ROC-AUC. ROC-AUC means Area Under the Receiver Operating Characteristic Curve.

## 3.4 Experiments

This section explains the experiments performed on the PolitiFact and PHEME datasets using all the six balancing techniques, as shown below.

**PolitiFact dataset experiments.** The following table illustrates the best results for the PolitiFact dataset with 10 epochs for each balancing and the performance measures for each balancing (Table 1).

**PHEME dataset experiments.** The table below clarifies the best results on the PHEME dataset with 10, 20, 50, and 100 epochs for each balancing technique and the performance measures for each one (Table 2).

**Table 1** PolitiFact dataset experiments

Balancing technique	Acc.	Pre.	Rec.	F1.	ROC
RandomOverSampler	0.95	0.94	0.95	0.94	0.95
SMOTE	0.95	0.94	0.93	0.94	0.93
RandomUnderSampler	0.96	0.95	0.94	0.94	0.94
NearMiss	0.91	0.87	0.91	0.89	0.91
CombineSMOTE UnderSampling	0.96	0.95	0.93	0.94	0.93
CombineOverUnderSampling	<b>0.96</b>	<b>0.96</b>	<b>0.95</b>	<b>0.95</b>	<b>0.95</b>

**Table 2** PHEME dataset experiments

Data set	Epo No.	Acc.	Pre.	Rec.	F1.	ROC
RandomOverSampler	10	0.82	0.81	0.80	0.80	0.80
	20	0.82	0.80	0.82	0.81	0.82
	50	0.85	0.84	0.84	0.84	0.84
	100	0.86	0.84	0.85	0.85	0.84
SMOTE	10	0.79	0.79	0.76	0.77	0.76
	20	0.80	0.79	0.78	0.78	0.78
	50	0.83	0.82	0.81	0.82	0.82
	100	0.82	0.80	0.80	0.80	0.80
RandomUnderSampler	10	0.80	0.82	0.78	0.79	0.79
	20	0.82	0.81	0.80	0.81	0.80
	50	0.83	0.82	0.84	0.83	0.84
	100	0.84	0.82	0.83	0.83	0.83
NearMiss	10	0.69	0.69	0.71	0.69	0.71
	20	0.77	0.76	0.77	0.76	0.77
	50	0.78	0.78	0.80	0.78	0.81
	100	0.81	0.80	0.81	0.80	0.81
CombineSMOTEUnderSampling	10	0.82	0.80	0.80	0.80	0.80
	20	0.82	0.80	0.81	0.81	0.82
	50	0.85	0.84	0.83	0.83	0.83
	100	0.83	0.82	0.83	0.82	0.83
CombineOverUndeSampling	10	0.81	0.80	0.78	0.79	0.78
	20	0.84	0.83	0.84	0.83	0.84
	50	0.86	0.85	0.84	0.85	0.84
	100	<b>0.87</b>	<b>0.86</b>	<b>0.85</b>	<b>0.85</b>	<b>0.85</b>

## 4 Results and Analysis

This section analyzes the performance and results of the proposed models for the PolitiFact and PHEME datasets.

**PolitiFact dataset.** The table “PolitiFact dataset experiments” shows that the balancing techniques (combination of SMOTE and RandomUnderSampler and the combination of RandomOverSampler and RandomUnderSampler) achieve the best results compared to the other regular balancing techniques. Moreover, the latter technique performs the best in the performance measurements shown.

Therefore, this experiment was repeated 30 times and the average of the performance measures was calculated to ensure the stability of the results of this experiment, where it showed high stability (see Table 3).

*Compare the best result with the related works.* The table below clarifies the compression between the best result and some related works that work on the same dataset. Our model is better than most related work; the paper [6] got approximate results (Table 4).

**PHEME dataset.** As with the first dataset, the balancing techniques (combination of SMOTE and RandomUnderSampler and the combination of RandomOverSampler and RandomUnderSampler) achieved the best results compared to the other regular

**Table 3** PolitiFact dataset stability

Stability	Acc.	Pre.	Rec.	F1.	ROC
Best result	0.96	0.96	0.95	0.95	0.95
Average of (30 results)	0.96	0.947	0.945	0.947	0.943

**Table 4** Compare the best result of the PolitiFact dataset with the related works

Source	Model	PolitiFact dataset				
		Acc.	Pre.	Rec.	F1.	AUC
[17]	Social article fusion	0.691	0.638	0.789	0.706	
[5]	Base	0.80	0.79	0.78	0.78	
	Hybrid	0.77	0.76	0.76	0.76	
[4]	(Text + Image)	0.846				
[6]	SFFN with MCDropout	<b>0.9115</b>	<b>0.9115</b>	<b>0.9115</b>	<b>0.9115</b>	
[18]	XLNet	0.895			0.90	
	TM	0.871			0.901	
[19]	FakeFlow	0.86	0.86	0.86	0.85	
	CombineOverUnderSampling	<b>0.96</b>	<b>0.96</b>	<b>0.95</b>	<b>0.95</b>	<b>0.95</b>

**Table 5** PHEME dataset stability for 100 epochs

Stability	Acc.	Rec.	Pre.	F1.	ROC
Best result	0.87	0.86	0.85	0.85	0.85
Average of (30 results)	0.848	0.832	0.835	0.836	0.835

balancing techniques. Moreover, the latter method performs best in terms of the performance measures shown. As the number of epochs increases, the performance results also improve. As shown in the table of experiments in the PHEME dataset, the results of 100 epochs reached 0.87 for the last balancing techniques, while they did not reach more than 0.82 for the 10 epochs. Accordingly, the experiment was repeated 30 times and the average of the performance measures was calculated to ensure the stability of the results of the experiment, showing high stability (see Table 5).

*Compare the best result with the related works.* The table below illustrates the comparison between the best result and some related works using the same dataset. Our model is better than all related works, except for the CNN + IG-ACO NB model for the Ferguson unrest event which achieved a precision of 0.874 and an F1-score of 0.857 in Kumar et al. experiment [9]. However, the result for all events still needs to be improved (Table 6).

Also note that the proposed model is more suitable for big datasets (such as the PolitiFact dataset) than for small datasets (such as the PHEME dataset), where the accuracy reached 0.96 in the PolitiFact dataset, while it did not reach more than 0.87 in the PHEME dataset.

The following section is the final section. It contains the conclusion of this work and the future work points.

## 5 Conclusion and Future Work

This work presents a fake tweet detection model that uses a deep learning algorithm (Bi-LSTM) and a combination of two balancing techniques RandomOverSampler and RandomUnderSampler. This model includes two merged models, one for textual attributes and the other for numeric/categorical attributes. This is a feature where most researchers have applied their work to textual attributes only. The model was trained on two datasets, the PolitiFact dataset and the PHEME dataset. The experimental results are very satisfactory and outperform many related works.

In the future, look forward to working with more datasets than those used in this work in the future to replicate the model, such as PHEME for five events, Twitter 15 and Twitter 16 datasets.

Train with many algorithms and compare them, such as LSTM and BERT. In addition, try changing the number of input layers, hidden layers, weights, and error rate and compare the results.

**Table 6** Compare the best result of the PHEME dataset with the related works

Source	Model	PHEME dataset					
		Acc.	Pre.	Rec.	F1.	AUC	
[9]	CNN + IG-ACO NB model	Germanwings crash event		0.767	0.761	0.763	0.76
		Charlie Hebdo event		0.856	<b>0.841</b>	0.848	0.90
		Ottawa shooting event		0.749	0.801	0.773	0.75
		Sydney Siege event		0.740	0.699	0.719	0.76
		Ferguson unrest event		<b>0.874</b>	<b>0.841</b>	<b>0.857</b>	<b>0.84</b>
		Overall events		0.776	0.745	0.732	
[8]	MTL3	Germanwings crash event	0.420			0.429	
		Charlie Hebdo event	0.369			0.327	
		Ottawa shooting event	0.645			0.352	
		Sydney Siege event	0.575			0.350	
		Ferguson unrest event	0.338			0.189	
[20]	StA-HiTPLAN + Time delay					0.774	
[10]	GCN	Charlie	0.790	0.790	0.790	0.864	0.690
		German	0.715	0.717	0.716	0.709	0.720
		Ottawa	0.675	0.681	0.677	0.655	0.680
		Sydney	0.655	0.655	0.664	0.690	0.660
		Ferguson	0.705	0.671	0.658	0.783	0.660
[21]	NLI-SAN (Rumor + Evidence)	Germanwings crash event				0.365	
		Charlie Hebdo event				0.354	
		Ottawa shooting event				0.591	
		Sydney Siege event				0.458	
		Ferguson unrest event				0.256	
[22]	RootText + PSA (MEAN)	0.463				0.416	
	CombineOverSamplerUnderSampling	<b>0.87</b>	<b>0.86</b>	<b>0.85</b>	<b>0.85</b>	<b>0.85</b>	

## References

1. Statista Research Department, “Global social media ranking 2019. Statista. <https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/>. Accessed 07 Aug 2022

2. Lua A (2022) 21 Top social media sites to consider for your brand. Buffer library. <https://buffer.com/library/social-media-sites/>. Accessed 20 Mar 2022
3. Sayce D (2022) The number of tweets per day in 2020. David Sayce. <https://www.dsayce.com/social-media/tweets-day/>. Accessed 20 Mar 2022
4. Singhal S, Kabra A, Sharma M, Shah RR, Chakraborty T, Kumaraguru P (2020) SpotFake+: a multimodal framework for fake news detection via transfer learning (student abstract). AAAI 34(10), Art. no. 10. <https://doi.org/10.1609/aaai.v34i10.7230>
5. Oriola O (2020) Exploring N-gram, word embedding, and topic models for content-based fake news detection in fakenewsnet evaluation. Int J Comput Appl 176:24–29. <https://doi.org/10.5120/ijca2020920503>
6. Das SD, Basak A, Dutta S (2021) A heuristic-driven uncertainty based ensemble framework for fake news detection in tweets and news articles. [arXiv:2104.01791](https://arxiv.org/abs/2104.01791) [cs]. Accessed 20 Apr 2022. [Online]. Available: [http://arxiv.org/abs/2104.01791](https://arxiv.org/abs/2104.01791)
7. Last F, Douzas G, Bacao F (2018) Oversampling for imbalanced learning based on K-Means and SMOTE. Inf Sci 465:1–20. <https://doi.org/10.1016/j.ins.2018.06.056>
8. Kochkina E, Liakata M, Zubia A (2018) All-in-one: multi-task learning for rumour verification. In: Proceedings of the 27th international conference on computational linguistics. Association for Computational Linguistics, Santa Fe, New Mexico, USA, pp 3402–3413. [Online]. Available: <https://aclanthology.org/C18-1288>
9. Kumar A, Bhatia MPS, Sangwan SR (2022) Rumour detection using deep learning and filter-wrapper feature selection in benchmark Twitter dataset. Multimed Tools Appl 81(24):34615–34632. <https://doi.org/10.1007/s11042-021-11340-x>
10. Sharma S, Sharma R (2020) Identifying possible rumor spreaders on Twitter: a weak supervised learning approach. CoRR abs/2010.07647, [Online]. Available: <https://arxiv.org/abs/2010.07647>
11. Shu K (2022) KaiDMML/FakeNewsNet. Accessed 06 Apr 2022. [Online]. Available: <https://github.com/KaiDMML/FakeNewsNet>
12. Shu K et al (2021) Leveraging multi-source weak social supervision for early detection of fake news. [arXiv:2004.01732](https://arxiv.org/abs/2004.01732) [cs, stat]. Accessed 25 Mar 2021. [Online]. Available: [http://arxiv.org/abs/2004.01732](https://arxiv.org/abs/2004.01732)
13. Poynter Institute, “PolitiFact.” <https://www.politifact.com/>. Accessed 27 Feb 2023
14. Zubia A, Hoi GWS, Liakata M, Procter R, Tolmie P (2015) Analysing how people orient to and spread rumours in social media by looking at conversational threads. CoRR abs/1511.07487, [Online]. Available: [http://arxiv.org/abs/1511.07487](https://arxiv.org/abs/1511.07487)
15. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
16. Schmidhuber J (2015) Deep learning in neural networks: an overview. Neural Netw 61:85–117. <https://doi.org/10.1016/j.neunet.2014.09.003>
17. Shu K, Mahadeswaran D, Wang S, Lee D, Liu H (2020) FakeNewsNet: a data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big Data 8(3):171–188. <https://doi.org/10.1089/big.2020.0062>
18. Bhattacharai B, Grammo OC, Jiao L (2021) Explainable Tsetlin Machine framework for fake news detection with credibility score assessment. In: International conference on language resources and evaluation
19. Ghanem B, Ponzetto SP, Rosso P, Pardo FMR (2021) FakeFlow: fake news detection by modeling the flow of affective information. CoRR abs/2101.09810, [Online]. Available: <https://arxiv.org/abs/2101.09810>
20. Khoo LMS, Chieu HL, Qian Z, Jiang J (2020) Interpretable rumor detection in microblogs by attending to user interactions. CoRR abs/2001.10667, [Online]. Available: <https://arxiv.org/abs/2001.10667>
21. Dougrez-Lewis J, Kochkina E, Arana-Catania M, Liakata M, He Y (2022) PHEMEPlus: enriching social media rumour verification with external evidence. In: Proceedings of the fifth fact extraction and verification workshop (FEVER). Association for Computational Linguistics, Dublin, Ireland, pp 49–58. <https://doi.org/10.18653/v1/2022.fever-1.6>

22. Wu J, Hooi B (2022) Probing spurious correlations in popular event-based rumor detection benchmarks. arXiv preprint [arXiv:2209.08799](https://arxiv.org/abs/2209.08799)

# An Intelligent Agent Framework for Resilient Deployment in the Internet of Things Environment



Nainsi Soni and Saurabh Kumar

**Abstract** In the last few years, the use of technology for post-disaster management is encouraged on different platforms. The technological advancements have provided an opportunity to utilize the automated and real-time communication and computation. The evolution of the Internet of Things has played a vital role in sensing and communication of the sensed activities from anywhere in the world. However, the automated deployment of devices is one of the major concerns, especially with the heterogeneous characteristics of intelligent devices and ever changing environmental conditions. Additionally, there is a need to model the intelligent entities existing in the IoT environment under an umbrella framework, so that the cooperation and coordination can emerge from the operation of these devices in the network. In this context, this paper proposes an agent-based cooperative intelligent framework to map the operation of devices in the IoT environment. Furthermore, an  $N$ -level agent-based deployment algorithm is proposed which utilizes the collective intelligence to provide optimized coverage in the region of interest. The proposal is evaluated on the IoT-based platform to test its efficacy.

**Keywords** Internet of Things · Multiagent systems · Deployment

## 1 Introduction

The last three decades have observed numerous disaster incidents occurring around the world. According to the IFRC world disaster report [1], the loss of human lives is estimated to be more than 3 crores, only due to natural disasters. Additionally, there are incidents reported for man-made disasters as well. In the year 2021, there were 432 disasters reported with 10, 492 deaths and an economic damage of nearly \$252

---

N. Soni (✉) · S. Kumar

Department of Computer Science and Engineering, The LNM Institute of Information Technology, Jaipur, Rajasthan, India  
e-mail: [20pcs002@lnmiit.ac.in](mailto:20pcs002@lnmiit.ac.in)

S. Kumar

e-mail: [saurabh.kumar@lnmiit.ac.in](mailto:saurabh.kumar@lnmiit.ac.in)

billion [2]. To cater to the demand of disaster management, the different agencies around the world are working to predict and proactively handle these disasters. In this context, the role of technology is significant, as an enabling technique, for the disaster management authorities.

The technological progress, especially in the form of ubiquitous computing [3], has played a crucial role in prediction and dissemination of information related to the disasters to anywhere and at any time. In its nascent form, known as Internet of Things (IoT), the ubiquitous computing provides a platform where the support of device-to-device communication helps in real-time communication and computation [4]. By considering the disaster as a sudden disruptive event caused by nature in a particular area, it becomes crucial to assess the IoT environment with respect to its capability. In its current stage, it provides a platform where heterogeneous devices co-exist and perform their respective operations. Moreover, there is a need to synchronize the capabilities of these devices to help the agencies of disaster management to perform efficient operation. It is, therefore, expected that the devices must implement the ambient intelligence at different levels to help in various disaster situations, especially when these devices are also affected due to disasters. Thus, the deployment of these intelligent devices becomes a critical challenge.

The deployment of devices in IoT environment poses a few challenges [5]. First, the system entities are distributed. In such a scenario, it is required to model these entities in a framework. This may help to manage the heterogeneity, in terms of the type of devices and their operational characteristics. Second, the automated operation is expected from these devices for real-time communication and computation. It requires the implementation of intelligent decision making at the different levels of operation. Third, there is a need to handle the interoperability among the devices, which further requires the analysis of various algorithms for accessibility of information. Finally, the deployment algorithm must be reliable enough to aid in cooperative information dissemination [6], as and when required.

There are different deployment techniques based on placement strategy, usage, surrounding, and the characteristics of nodes. Based on the placement strategy, the technique can be divided into random, deterministic, and quasi-random [7]. Based on usage, the techniques are either barrier-oriented or target-oriented. Similarly, the deployment gets affected by the environment characteristics such as open areas or indoor situations [8]. Moreover, the characteristics of nodes, in terms of homogeneous or heterogeneous operation, affects the deployment potential. However, it must be noted that the deployment algorithms work in three different phases, namely, pre-deployment, post-deployment, and re-deployment [9]. In all these phases, the devices may get destroyed by the situations of disaster. It is expected that these deployed devices must provide the services, irrespective of the adverse disaster situations. Thus, there is a need to implement the ambient intelligence among the devices so that they can contribute to the efficient operation after deployment.

The ambient intelligence can be achieved if the devices learn from the environment where they are deployed and being operational. By understanding that the challenges of disaster need integrated solution for both the disaster management systems and relevant authorities, the deployed devices need to collect spatial and temporal

information in real-time situations. Moreover, it is possible that a few of the devices stop functioning due to unforeseen circumstances. In such a scenario, there is a need to assign the specific roles and responsibilities to these devices, which may be replicated or reassigned to some other devices, as and when the problem occurs. This kind of resilience in the deployment technique can be implemented with intelligent decision-making strategies to communicate timely among the deployed entities.

To address the above-mentioned challenges and provide a resilient deployment in the IoT environment, this paper proposes an intelligent agent framework, which maps the different intelligent agents operating in the environment to work cooperatively. Further, the proposed work implements a collaborative deployment algorithm and tests its efficacy in the proposed framework. The novelty of the proposal can be outlined by its analysis of the connectivity potentials among the deployed devices using the proposed collaborative algorithm. The remainder of the paper is organized as follows. Section 2 presents the proposed intelligent agent framework, followed by discussion of proposed collaborative deployment algorithm in Sect. 3. The results are discussed in Sect. 4. Finally, Sect. 5 concludes the work.

## 2 Intelligent Agent Framework

In this section, the proposed framework named Collaborative Intelligent Network (CIN) for intelligent agents operating in the IoT environment is presented. The proposed framework aids in three important aspects. First, it assumes that all the devices operating in the IoT environment are agents. Essentially, the devices existing in the IoT network can be categorized into sensor and IoT devices [10]. Second, the agents operating in the network are modeled to implement a certain form of cooperation and coordination among themselves. Third, these agents are expected to achieve ambient intelligence to help in real-time communication and computation. The network, thus formed, is expected to have intelligent agents which perform collaborative processing to achieve both the network-oriented and Quality of Service (QoS)-based parametric efficiencies.

The IoT environment is assumed to be in any of the finite set of discrete and instantaneous states,  $S = \{s_1, s_2, s_3, \dots\}$ . The discreteness of states is a modeling assumption and signify that any continuous environment can be modeled by the discrete environment based on the expected degree of accuracy. An agent is assumed as any entity that senses the environment and perform the actuation to complete a desirable set of tasks. The environment initiates in one among the possible set of states,  $S$ . The agents are assumed to either have possible set of actions, or devise the actions based on the real-time situation. These agents may operate from anywhere, at anytime, and assumed to abide by the communication protocols supported by IoT. The agents are further assumed to be heterogeneous in its operational and manufacturing characteristics.

An agent must operate based on the requirements of the environment in which it exists. Let us assume that  $A = \{a_1, a_2, a_3, \dots\}$  denotes the set of actions which an

agent takes to transform the state of the environment. The actions are performed in incremental pattern. For instance, an agent existing in state  $s_1$  takes an action  $a_1$ , due to which the state changes to  $s_2$ . At state  $s_2$ , the agent again chooses an action based on which the further changes in the states will be possible. Since there are numerous such agents operating in the IoT environment, each agent will have different sets of actions and associated change of states occurring due to the actions taken. Thus, an agent will have a *pass*, which consists of a sequence of interleaved pairs of states and associated actions. Let  $P = \{p_1, p_2, p_3, \dots\}$  be a set of such passes for an agent, i.e., a set of all possible finite sequences over  $S$  and  $A$ . Thus, it is further assumed that  $P^A \subset P$  ends with an action, and  $P^S \subset P$  ends with an environment state.

Moreover, the various sensors and IoT devices may have different effects of their actions on the environment in which they exist. A state reformer function is used to represent this effect of agent's actions on the environment. It maps a pass of an agent to a set of possible states of the environment and is given by

$$\tau : R^A \rightarrow \rho(S) \quad (1)$$

where a *pass* is assumed to end with the action of any agent. Further, the sensors and IoT devices in the environment must adhere to the historical data to take actions. Thus, it is also assumed that the environment characteristics are both history dependent and non-deterministic. The environment is defined as a triple given by  $E = \{S, s_1, \tau\}$ , where  $s_1$  is the initial state of the environment. Similarly, an agent can be modeled as a function mapping the passes to actions, wherein it is assumed that the passes must end with an environmental state such that

$$Ag : R^S \rightarrow A \quad (2)$$

where  $Ag = \{Ag_1, Ag_2, \dots\}$  is the set of all agents existing in the environment. Since the deployment of agents is costly, the challenge is to provide coverage of the region with optimum number of agents being deployed. In such a case, the proposed framework assumes that the environment may be non-deterministic implicitly, but the agents are assumed to be deterministic. Thus, the proposed CIN framework assumes a system being a pair of agent and environment with possible set of *passes* given by  $P(Ag, E)$ .

Furthermore, the agents are assumed to have a utility function,  $U$  that provides a map from the states of the environment to a real value. This real value indicates the reward that an agent gets on reaching a particular state. The aim of an agent is to gain maximum rewards. Also, since the agents are involved in cooperation and coordination among themselves, it is assumed that an agent must think about the maximization of its own rewards as well as the rewards of the society in which it exists. The proposed CIN framework focuses on the utilization of collective intelligence to operate in the IoT environment. The deployment algorithm for agents utilizing the proposed framework is discussed in the following section.

### 3 Proposed Deployment Algorithm

In this section, the proposed algorithm for deployment of agents using the proposed CIN framework in the IoT environment is discussed. Generally, the deployment works in three phases: pre-deployment, post-deployment, and re-deployment. The main aim of the proposed deployment technique is to address the pre-deployment phase. In pre-deployment phase, the agents are initially placed in the environment. In the IoT environment, there is an existence of huge number of heterogeneous devices with different ranges of coverage. In this context, these devices, acting as agents, are initially categorized into 0-level, 1-level, and 2-level agents. The 0-level agent does not recognize the existence of other agents in the environment. Similarly, the 1-level agent recognizes the other agents where actions affect the reward of agent. On the other hand, the 2-level agent believes that all other agents are at level 1. The proposal creates this categorization based on the areas of coverage of the agents. It is assumed that the area of coverage is based on the communication range,  $r$  of an agent. The agent lies at the center of the circle with radius,  $r$ .

Further, the proposed deployment techniques uses incremental approach to place the agents on the terrain, i.e., the agents are deployed based on the collective intelligence of the already deployed agents in the environment. In this technique, the deployment is performed using the functionalities provides by the three level hierarchy of agents. There are two choices available for the agents. First, agents learn to correlate their actions with the rewards. Second, the agents learn to predict the actions of other agents and utilize this knowledge for its own domain. Since the terrain environment in the proposal is assumed to be non-deterministic, it is possible that the coverage area of the terrain is unknown. In this context, there is a need to deploy the agents optimally considering the cost of deployment. Also, the proposal uses a mix of deterministic and random deployment strategies to cater to the non-deterministic terrain requirements.

Initially, suppose that there are  $n$  agents available for deployment. The algorithm chooses  $k \subset n$  number of agents to be deployed randomly on the terrain. The environment sensing of these  $k$  agents will be independent of each other, and thus, are considered as 0-level agents. Their task is to sense the activities in their respective coverage ranges with an aim to increase the rewards,  $U = \{u_1, u_2, \dots, u_k\}$ . The reward is given based on the accuracy of sensing in respective coverage range. It may be possible that there are some regions on the terrain which will be not covered by 0-level agents, known as *gap spots*. In such a case, the next increment of deployment is needed.

Due to the *gap spots*, it is possible that the terrain has coverage holes. To cater to these coverage holes and to fill the gaps, the proposed algorithm uses quasi-random deployment [9] technique to deploy  $t \subset (n - k)$  agents which uses a mix of deterministic and random deployment methodology. The agents, thus deployed, form the 1-level ecosystem, wherein all the agents must have the knowledge of each other through probabilistic modeling of rewards. The probabilistic modeling of rewards will help in two aspects. First, it will help the agents in maximizing their

sensing in their respective coverage areas to get more rewards. Second, it will help in implementing the cooperative motivating effect of one agent's reward on another agent's functioning at the same level. It aligns each agent's reward function with the global reward and is quantified using

$$C_i(s, \vec{a}) = \sum_{\vec{a}' \in \vec{A}} \frac{\theta[u_i(s, \vec{a}') - u_i(s, \vec{a})]}{|\vec{A}|} \quad (3)$$

where,  $C_i((s, \vec{a}))$  is the preference of  $i$ th agent over  $((s, \vec{a})$  and  $\theta(y)$  is a unit step function defined as

$$\theta(y) = \begin{cases} 1, & y \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

Similarly, the global preference is defined as

$$C(s, \vec{a}) = \sum_{\vec{a}' \in \vec{A}} \frac{\theta[U(s, \vec{a}') - U(s, \vec{a})]}{|\vec{A}|} \quad (5)$$

It must be mentioned that the target of an agent is to match its reward with the global preference of the whole system in which it operates. This can only be achieved when the agent maximizes the sensing operation in its own vicinity and knows about its contribution to the global reward of the 1-level hierarchy. In this way, the 1-level agents provides the support, in terms of coverage, to the 0-level agents in the hierarchy. However, it is still possible that during the initial deployment phase, the terrain parameters change due to the external factors unbeknownst to the agents. In such a scenario, it becomes important to cater to the coverage holes further in the network.

The proposed algorithm implements  $n - (k - t)$  number of agents in the environment which forms the 2-level agent society. An agent deployed at this level assumes all other agents at level 1 and tries to find the reward function that has low opacity and highly factored. A factored system is more likely to converge to the set of policies that maximizes the reward. On the other hand, opacity defines that an agent's reward has impact on another agent as the target function may change. In this case, the change in the terrain parameters may affect the achievement of the optimum deployment with maximized coverage of the region. Thus, an agent at 2-level tries to find the probability with which the agents at level 0 and 1 wanted to sense a region but could not. It measures the difference in the global utility between the agent's action and its expected action and is given by

$$U_i(s, \vec{a}) = U(s, \vec{a}) - \sum_{\vec{a}' \in \vec{A}} Pr(\vec{a}') U(s, \vec{a}_{-i}, \vec{a}') \quad (6)$$

---

**Algorithm 1** Agent-based Resilient Deployment Algorithm

---

**Require:**  $n$  agents,  $U$  set of rewards for agents,  $S$  environment states,  $A$  set of actions.

**Ensure:** Coverage of environment with optimal number of agents.

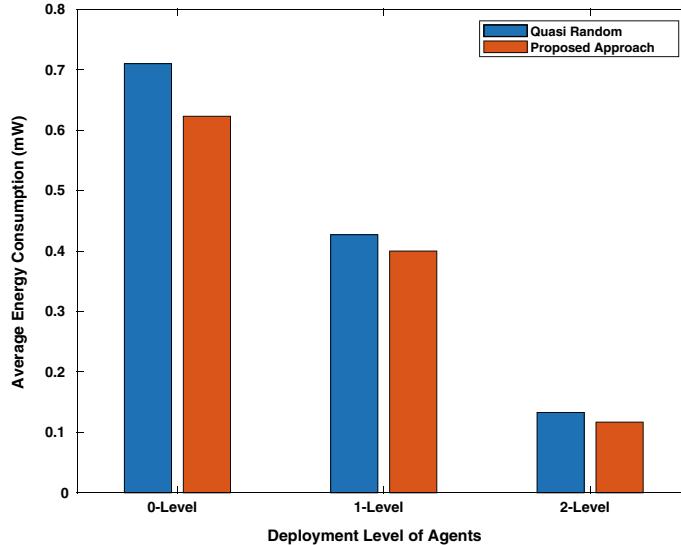
- 1: Define a terrain  $T$ .
- 2: Deploy  $k \subset n$  0-level agents on  $T$  randomly.
- 3: **if**  $gap\ spots$  is observed **then**
- 4: Deploy  $t \subset (n - k)$  1-level agents in  $gap\ spots$  using quasi-random deployment technique.
- 5: Calculate the agent's reward function using equation 3 and compare with global reward preference.
- 6: **if**  $gap\ spots$  is observed **then**
- 7: Deploy  $n - (k - t)$  2-level agents on  $T$  using quasi-random deployment technique.
- 8: Measure the global reward between agent's action and its expected action.
- 9: **if** global reward is achieved **then**
- 10: Stop.
- 11: **else**
- 12: Deploy more number of agents in the  $gap\ spots$ .
- 13: **end if**
- 14: **end if**
- 15: **end if**

---

where  $Pr(\vec{a'})$  is the probability of occurrence of  $\vec{a'}$ . If still the  $gap\ spots$  are found, repeat the process with more number of agents. The proposed deployment algorithm is summarized in Algorithm 1. The novelty of the proposed algorithm can be outlined by three important aspects. First, it uses the cooperative intelligent framework of agents to implement the collective intelligence, which helps in the placement of optimum number of agents in the network. Second, it implements an  $N$ -level agent-based deployment algorithm which uses a mixture of random and quasi-random deployment approaches. This helps in maintaining the even deployment in the terrain. Finally, it uses the probabilistic modeling to know about the existence of the other agents and their rewards, which helps in implementing the cooperative intelligence among the agents in the network. The results of implementation of the proposed algorithm is discussed in the following section.

## 4 Results and Discussion

In this section, the implementation of the proposed resilient deployment algorithm using the CIN framework is discussed. The implementation of the proposal is done in two steps. In the first step, the  $N$ -level agent-based deployment is performed on the Python platform. In second step, the proposal is tested on IoT-based Cooja emulation engine [11] to check its efficacy using IoT devices. Initially, the agents are deployed on a  $50 \times 50$  terrain configured on the Python platform. The reward of an agents is assumed to be the minimized energy consumption during the operation. The reward is initially assumed to be 0. The global preference of reward is assumed to be 10 for 0-level agents with an estimated time limit of 5 minutes from the actual time



**Fig. 1** Average energy consumption by agents in different levels of society

of deployment of an agent. Incrementally, the deployment is performed for 1-level and 2-level agents in the region. For the terrain mentioned above, 32 devices are randomly deployed as 0-level agents, 17 and 11 devices are deployed respectively as 1-level and 2-level agents using the quasi-random technique. It must be noted that the deployment is performed based on the coverage *gap spots* in the region. The average energy consumption is calculated for all three levels of deployment. As shown in Fig. 1, it can be observed that at the higher levels of agent deployment, the average energy consumption reduces. This is due to the reason that the higher level agents utilize probabilistic modeling of the neighbors, and thus, know the expected rewards of itself and the other agents in the environment. Further, the connectivity and reachability of the network is computed for each agent. For  $n$  agents with degree  $\deg(n_d)$ , the value of connectivity  $\text{con}$  is defined as [12]

$$\text{con} = \sum_{i=1}^{|n|} \frac{\deg(n_d)}{2 * \binom{|n|}{2}} \quad (7)$$

It must be noted that the full connectivity can be achieved with value of  $\text{con} = 1$ . Similarly, the reachability defines the level of indirectness while routing in the network. For  $n$  agents, where  $n$  are formed using  $q$  partitions  $q = \{q_1, q_2, \dots\}$ , the reachability  $\text{rch}$  is defined as [12]

$$\text{rch} = \sum_{l=1}^{|q|} \frac{\binom{|q_l|}{2}}{\binom{|n|}{2}} \quad (8)$$

**Table 1** Comparison of average connectivity and reachability

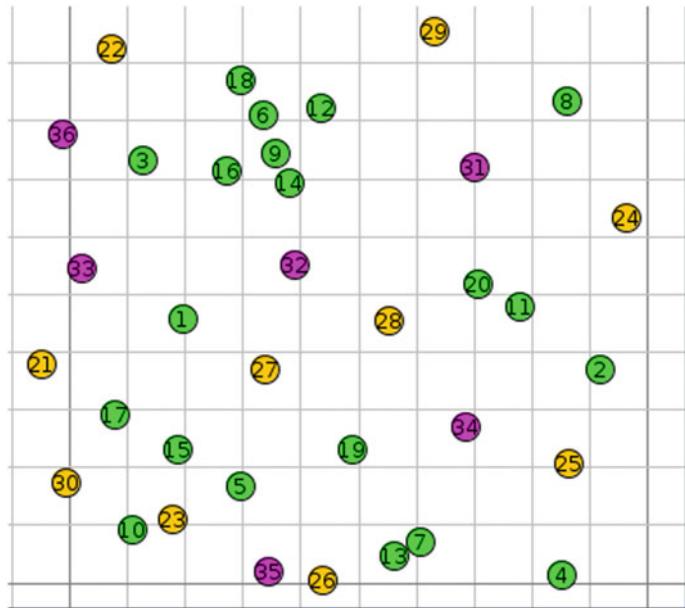
Parameter/ Algorithm	Quasi random approach		Proposed approach	
	Low density (%)	High density (%)	Low density (%)	High density (%)
Average connectivity	62.9	79.3	67.1	81.13
Average reachability	78.43	83.73	86.44	89.11

where the full reachability is achieved with  $r_{ch} = 1$ . Table 1 presents the results obtained for connectivity and reachability values for different densities of network. Here, the density of agents is varied from 10 to 60. Also, the degree of nodes is varied from 1 through 6. It can be observed that the average connectivity and average reachability is more for the proposed approach as compared to the quasi-random approach. One of the basic reasons is that the proposed approach randomizes the already randomized quasi-based deployment algorithm. This helps in further implementing the even distribution of agents in the region of interest.

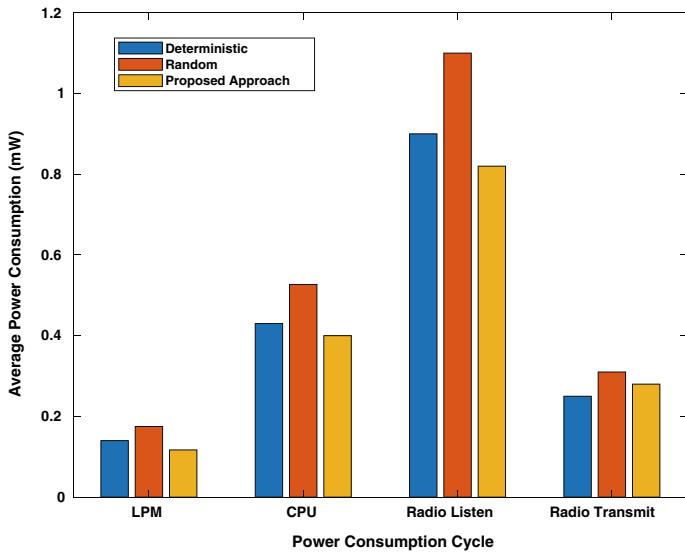
The proposed algorithm is further simulated on IoT-based Cooja platform. A sample snapshot of deployment on a  $100 \text{ m}^2$  terrain is shown in Fig. 2. The devices 1 through 20 are 0-level agents, devices 21 through 30 are 1-level agents, and devices 31 through 36 are 2-level agents. For the purpose of simulation, a unit disk graph medium with distance loss is configured. The agent initialization delay is set as 2 minutes. Two different densities of deployment is considered for the same, namely, low density and high density. The pre-deployment of agents is performed using random, quasi-random, and proposed algorithmic approaches.

The comparative analysis is performed for four different power modes of the motes, namely, low power mode (LPM), processing mode (CPU), radio listen, and radio transmit modes. The comparison is depicted in Figs. 3 and 4, respectively. It can be observed that the maximum energy is consumed in radio listen mode and minimum in the case of LPM mode. However, in all the modes, the proposed algorithm performs better than the random and quasi-random approaches of deployment. A significant reason is that the quasi-random approach utilizes a mix of both the deterministic and random approaches, while the proposed approach uses both the random and quasi-random approaches at 0th, 1st and 2nd levels, respectively. It is also observed that there is less requirement of 2-level agents in comparison to the lower level agents. This is due to the reason that the *gap spots* are reduced by the implementation of incremental approach to deployment, in addition to the utilization of quasi-random approach.

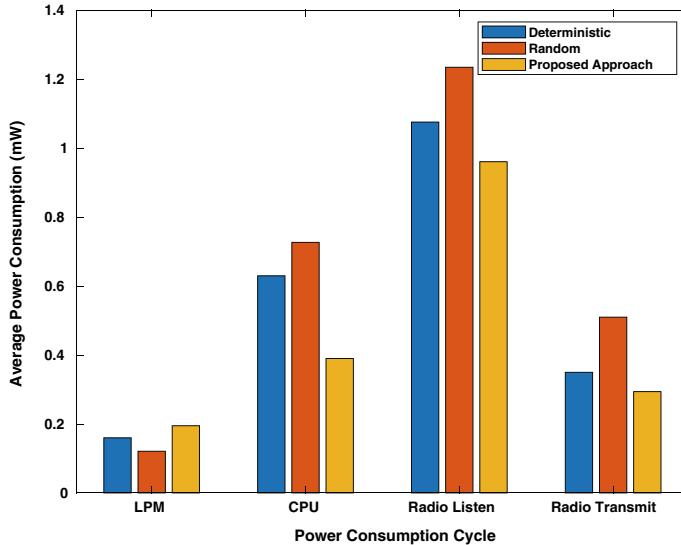
The pre-deployment is an important phase of the deployment, and thus, requires a careful implementation. This will further help in identification of the *gap spots* which are essential aspects to be considered during the post-deployment maintenance of these intelligent devices and their re-deployment, as and when required. In this context, the proposed cooperative intelligent network provides a platform for the intelligent agents to map themselves with respect to the intelligence levels



**Fig. 2** Deployment of 36 agents on  $100 \times 100$  m $^2$  terrain



**Fig. 3** Comparison of average power consumption under low density conditions



**Fig. 4** Comparison of average power consumption under high density conditions

required by different heterogeneous devices operating in the network. Furthermore, the proposed collaborative deployment algorithm with  $N$ -level hierarchy brings in the collective intelligence which may be helpful in implementing the learning models for deployment.

## 5 Conclusions

The post-disaster management is one of the main concerns of the world in current era. Due to the evolution of the IoT environment, it has become possible to communicate and compute the real-time data acquired from anywhere and at any time. However, the automated processing requires that the devices must be able to sense and actuate upon the environment with respect to improved coverage of the region. In such a scenario, the deployment must be done in such a way that the collective intelligence can be utilized to deploy these devices with optimum number of devices. In this context, this paper proposes a cooperative intelligent framework to map the devices as agents and intelligence levels of these agents to provide learning at these different levels. Further, the paper also proposes a resilient deployment algorithm using  $N$ -level agents. The proposed algorithm is implemented and compared with the existing quasi-random deployment scheme. It is found that the proposal provides efficient results with respect to the connectivity, reachability, and energy consumption. In the future, the proposal can be explored for the post-deployment and re-deployment phases to provide cooperative learning-based adaptive deployment of devices.

## References

1. Scott-Smith T (2018) Paradoxes of resilience: a review of the world disasters report 2016. *Develop Change* 49(2)
2. Hagon K (2021) Protecting people in the context of climate change and disasters: Setting the scene. In: Proceedings of the ASIL annual meeting, vol 115. Cambridge University Press, pp 156–158
3. Greenfield A (2010) Everyware: the dawning age of ubiquitous computing. New Riders
4. Mukhopadhyay SC, Suryadevara NK (2014) Internet of things: challenges and opportunities. Springer
5. Suresh P, Daniel JV, Parthasarathy V, Aswathy R (2014) A state of the art review on the internet of things (IoT) history, technology and fields of deployment. In: Proceedings of the International conference on science engineering and management research (ICSEMR). IEEE, pp 1–8
6. Van den Bergh F, Engelbrecht AP (2004) A cooperative approach to particle swarm optimization. *IEEE Trans Evol Comput* 8(3):225–239
7. Khoufi I, Minet P, Laouiti A, Mahfoudh S (2017) Survey of deployment algorithms in wireless sensor networks: coverage and connectivity issues and challenges. *Int J Auton Adapt Commun Syst* 10(4):341–390
8. Kulkarni N, Prasad R, Cornean H, Gupta N (2011) Performance evaluation of aodv, dsdv & dsr for quasi random deployment of sensor nodes in wireless sensor networks. In: Proceedings of the international conference on devices and communications (ICDeCom). IEEE, pp 1–5
9. Pandey SK, Zaveri MA (2016) Optimized deployment strategy for efficient utilization of the internet of things. In: Proceedings of the international conference on advances in electronics, communication and computer technology (ICAECCT). IEEE, pp 192–197
10. Pandey SK, Zaveri MA (2018) Quasi random deployment and localization in layered framework for the internet of things. *Comput J* 61(2):159–179
11. Osterlind F, Dunkels A, Eriksson J, Finne N, Voigt T (2006) Cross level sensor network simulation with cooja. In: Proceedings of the 31st IEEE conference on local computer networks. IEEE, pp 641–648
12. Chrobak M, Karloff H, Radzik T (1991) Connectivity versus reachability. *Inf Comput* 91(2):177–188

# The Impact of Financial Ratios and Pandemic on Firm Performance: An Indian Economic Study



Manpreet Kaur Khurana , Shweta Sharma , and Navneet Bhargava

**Abstract** This study aims to assess the impact of COVID-19 on company performance in various industries as defined by the Global Industry Classification Standard (GICS). A range of liquidity and financial variables, including net working capital, quick ratio, debt-equity ratio, and financial autonomy rate, are used in the study to assess the company's performance across various industries. The study uses trend analysis and panel regression techniques to investigate the relationship between liquidity, financial ratios, and firm performance. The findings from the study suggest that COVID-19 has a negative impact on the performance of companies in sectors such as communication services, consumer discretionary, financial, industrial, material, real estate, and information technology (IT). In contradiction, the healthcare and energy industries both reported a positive relationship. Furthermore, regression analysis demonstrates a positive correlation between debt financing, working capital management (WCM), and firm performance. In contrast, the relationship between financial autonomy and firm performance is negative. Our findings can assist policy-makers, such as shareholders, investors, and stockholders, in determining the optimal decisions for management and investment activities.

**Keywords** Financial ratios · COVID-19 · Firm performance · Liquidity

## 1 Introduction

The COVID-19 pandemic has aggravated the problem of financial management. The financial crisis of COVID-19 was considerably worse than the financial crisis of 2008–2009 [1]. The pandemic had a detrimental effect on several industries and

---

M. K. Khurana · S. Sharma · N. Bhargava  
Malaviya National Institute of Technology, Jaipur, India  
e-mail: [shweta.dms@mnit.ac.in](mailto:shweta.dms@mnit.ac.in)

M. K. Khurana  
e-mail: [2020rbm9569@mnit.ac.in](mailto:2020rbm9569@mnit.ac.in)

sectors, including sales, operations, and marketing [1]. Additionally, the coronavirus pandemic substantially impacted the global financial system.

According to the Ministry of Statistics and Programme Implementation annual report (2021–22), the growth rate in India decreased by 3.1% in the fourth quarter of 2020<sup>1</sup>. The unemployment rate increased from 6.7% in March to 26% in April 2020. During the lockdown, 140 million people lost their jobs, and others had their pay cut in half. The Indian economy shrank about \$4.5 billion daily during the early shutdown phase (March 25–April 14, 2020). The COVID-19 pandemic hindered Indian exports and imports. In April 2020, the Ministry of Commerce and Industry, Government of India, released press information regarding imports and exports of India. The report revealed that imports and exports were 36.65% and 47.36% lower in April 2020 than last year in India. The export of gems and jewelry decreased to 98.74% in April 2020, leather and leather goods fell to 93.28%, and handicrafts and ceramics exports decreased to 91.84% and 91.67%, respectively<sup>2</sup>. The valuation of the retail sector in India remained at \$790 billion in 2019. In recent years, electronic commerce has flourished, and market forecasts foresee an increase of 30% in online trade by 2020 (National Investment Promotion and Facilitation Agency, 2020)<sup>3</sup>.

Several studies have investigated the impact of COVID-19 on the company's performance [2, 3]. The performance of Indian enterprises is negatively impacted by COVID-19, according to research by Shen et al. [4] based on a sample of Chinese companies between 2014 and 2020. To the author's knowledge, there needs to be more research focusing on the impact of financial ratios on firm performance.

In the first section of the study, the researcher uses a graphical presentation to show the trajectory of the financial performance of Indian firms following the COVID-19 epidemic. The study then examines how COVID-19 affects the firm's performance across 10 GICS industrial categories. The third stage involves analyzing the performance of businesses across industries using liquidity and financial ratios such as net working capital, quick ratio, debt-equity ratio, and financial autonomy rate.

The remainder of this research paper is structured as follows: In Sect. 2, we present the literature review. Section 3 will discuss the research methodology, sampling, and data collection techniques. Section 4 covers trend analysis utilizing financial ratios. Section 5 will conclude with relevant research findings, limitations, and future directions for research.

<sup>1</sup> <https://mospi.gov.in/documents/213904/1885585/Printed+Annual+Report+2021-22+%28Eng.%29.pdf>

<sup>2</sup> <https://pib.gov.in/PressReleasePage.aspx?PRID=1624102>

<sup>3</sup> <https://www.investindia.gov.in/sector/retail-e-commerce/e-commerce>

## 2 Literature Review and Hypotheses Development

### 2.1 Debt and Firm Performance

Most of the earlier research studies from developed countries suggest a favorable relationship between debt and firm performance [5, 6]. However, a few developing countries show a negative correlation between the two, such as Ghana, India, and Jordan [7]. Thus, from the above results and past literature review, we developed our first hypothesis:

H1: There is a negative relationship between debt-equity ratio and the firm performance.

### 2.2 Working Capital and Firm Performance

According to the literature, there is an inconsistency in the association among WCM and firm performance [8]. Using a sample of 263 Indian-listed firms, Sharma and Kumar [9] found a positive correlation between WCM and firm performance. Additionally, Shrivastava et al. [10] research found a poor correlation between firm performance and WCM. Using return of assets (ROA) and Tobin's Q as proxies for firm performance, Altaf and Ahmad [8] study of 437 Indian non-financial firms over ten years (2007–2016) portrays a positive association between WCM and firm performance. We propose a working hypothesis as follows.

H2: Efficient working capital management helps in the improvement of firm performance.

### 2.3 Financial Autonomy and Firm Performance

According to the literature review, there are many empirical studies on the association between retained earnings and firm performance. In a study of the top 50 Chinese e-commerce retailers, Yang and Wang [11] identified a negative correlation between financial autonomy and firm performance. However, Lokwang et al. [12] revealed a positive association between financial autonomy and firm performance on a sample of Kenyan companies. We offer the following working hypothesis:

H3: There is a negative relationship between financial autonomy and the firm performance.

## 2.4 Quick Ratio and Firm Performance

A higher quick ratio percentage could indicate that a company can pay off its current liabilities better. Borhan et al. [13] revealed a positive association between the quick ratio and the performance of manufacturing firms in India. Further, Yameen et al. [14], from a sample of pharmaceutical companies over ten years (2008–17), also found that quick ratio and performance are positively associated. Consistent with the literature, we hypothesize that:

H4: There is a positive relationship between quick ratio and the firm performance.

## 3 Sample

The final sample comprises 2280 firms listed on National Stock Exchange (NSE) in India. The panel dataset comprises 9058 observations from 2016 to 2021. According to the GICS framework, the organizations in our sample are classified into the following industrial sectors: Communication Services, Consumer Discretionary, Consumer Staples, Energy, Financials, Health Care, Industrials, IT, Material, and Real Estate.

## 4 Research Methodology

This study initially uses the pooled ordinary least squares (OLS) approach (baseline equation) to estimate the results. Nonetheless, we also employed fixed effect model (FEM) [15] and random effect model (REM) [16] regression techniques to address the heterogeneity and endogeneity issues [17].

$$\begin{aligned} \text{Firm Performance}_{it} = & \beta_0 + \beta_1 \text{Capital Structure}_{it} \\ & + \beta_2 \text{Working Capital}_{it} + e_{it} \end{aligned} \quad (1)$$

Here,  $i$  represents the company, and  $t$  represents the year.  $Y$  represents the firm's performance, as measured by return on assets (ROA) and return on equity (ROE).  $\beta_1$  refers to the estimated capital structure coefficient. The debt-to-equity ratio and financial autonomy rate are used in the equation to represent the capital structure in Eq. (1). Using this capital structure coefficient, we tested the null hypothesis that debt-equity ratio and financial autonomy rate have a negative impact on firm performance ( $Y$ ).  $\beta_2$  represents the estimated net working capital coefficient.

In Eq. (1), working capital is represented by net working capital and the quick ratio. Using this capital structure coefficient, we tested the null hypothesis that effective working capital management contributes to the enhancement of firm performance ( $Y$ ).

**Table 1** The table presents the description of key variables used in the study

Variable	Definition
<i>Dependent variables</i>	
Return on equity (ROE)	Earnings after interest and tax/Total shareholders' equity
Return on assets (ROA)	Earnings after interest and tax/Total assets
<i>Independent variables</i>	
Net working capital	Net working capital/Total assets
Cash ratio	(Cash + Marketable securities)/Current liabilities
Quick ratio	(Cash + Marketable securities + Account receivables)/Current liabilities
Receivables turnover ratio	Sales/Average account of receivables
Financial autonomy rate	Shareholders' equity/Total assets
Debt-equity ratio	Total debt/Shareholders' equity

Table 1 presents a detailed description of the variables considered in the study. We took accounting-based measures, such as ROA and ROE, as dependent variables. In addition, net working capital and quick ratio which signify liquidity management as independent variables. Furthermore, we took capital structure measures such as debt-to-equity ratio and financial autonomy rate as independent variables in our study.

## 5 Results

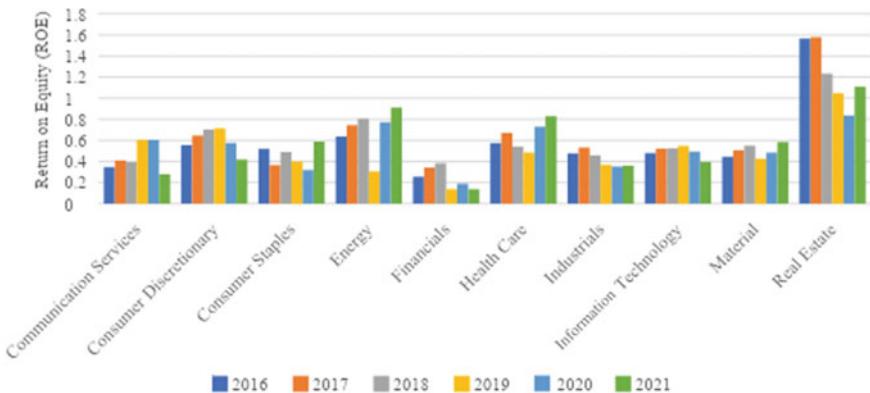
### 5.1 Trend Analysis

Figures 1 and 2 illustrate the trend in firm performance (using ROE and ROA as surrogate variables) before and after COVID-19. The impact of COVID-19 can be determined by analyzing the firm's performance in 2020–21. The figure represents a decline in firm performance across almost all industries. In support of this, PwC's COVID-19 pulse report presents a survey report offering a record prospectus from 2017 to 2022, demonstrating a decline in firm performance in the case of communication services due to a delay in 5G delivery and an increase in indebtedness caused by the COVID-19 outbreak<sup>4</sup>. Similar results were obtained from a survey, revealing an increase in electronic goods consumption [18].

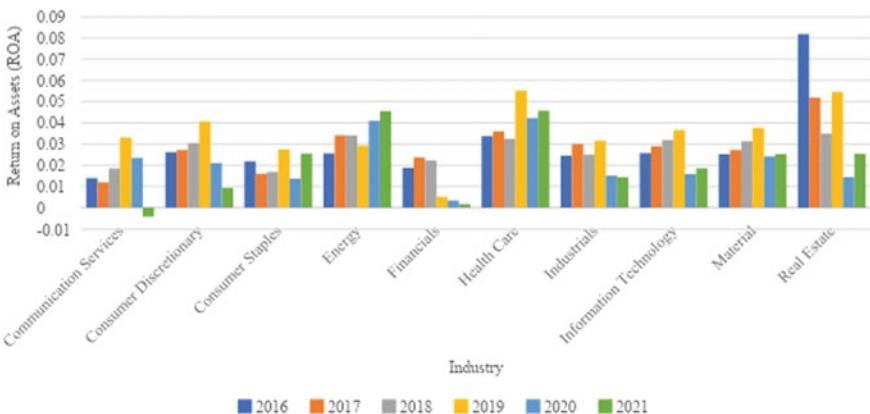
In addition, Figs. 1 and 2 reveal a decline in firm performance for consumer discretionary products, from 0.713 in 2019 to 0.575 in 2020 and then to 0.418 in 2021, as during the COVID-19 period and afterward, consumers attempted to spend more of their money on essential products, resulting in a decline in the demand for

---

<sup>4</sup> <https://www.pwc.com/us/en/library/covid-19/coronavirus-telecommunication-impact.html>



**Fig. 1** Trend analysis of ROE belonging to different sectors



**Fig. 2** Trend analysis of ROA from various sectors

non-essential items<sup>5</sup>. In support of Shen et al. [4] findings from a study conducted in China, Figs. 1 and 2 demonstrate a decline in the performance of the real estate business from 1.047 in 2019 to 0.836 in 2020. However, the situation began to improve beginning in 2021, when the real estate industrial sector's ROE increased to 1.109. Moreover, transportation disruptions, plant closures, and material scarcity have caused a decline in industrial and material sector performance [19].

Furthermore, Figs. 1 and 2 demonstrated a negative impact of COVID-19 on firm performance in the IT sector, which can be attributed to a prolonged lockdown and a dearth of end-user resources. In addition, contractual agreements were frequently

<sup>5</sup> [https://www.business-standard.com/article/economy-policy/india-seeing-consumption-revival-spending-on-discretionary-increasing-121092101043\\_1.html](https://www.business-standard.com/article/economy-policy/india-seeing-consumption-revival-spending-on-discretionary-increasing-121092101043_1.html)

settled based on discount offers, and there were delays in fulfilling contractual agreements (KPMG report 2020)<sup>6</sup>, resulting in a decline in the revenue generation of the Indian IT industry. COVID-19 significantly impacted the Indian financial market (insurance companies, banks, pension funds, and mutual funds). The risk of customer recovery and the moratorium extension have contributed to the decline in firm performance in the finance industry. According to a report published by KPMG in 2020, the healthcare sector has reported a significant increase in firm performance over the years, which can be attributed to business scaling and optimal utilization of resources.

Beyond this, the energy sector, which is composed of diversified segments like power sources ranging from conventional sources like coal, natural gas, oil, and natural gas and non-conventional sources like wind, solar, agriculture, and solar, demonstrates an increase in firm performance (Figs. 1 and 2) during 2020–21.

## 5.2 Regression Results

In Table 2, column (3) assesses the effect of the debt-to-equity ratio on the ROE of the companies. Similar to the findings from Berger and Di Patti [5] and Gill et al. [6], our results show a positive correlation between debt and firm performance, with a 4.969% increase in ROE for every percentage point increase in the debt–equity ratio. The findings are in contradiction to our stated premise (H1). In addition, the estimated relationship between net working capital and firm performance (ROE) is positive, with a coefficient of 0.97.  $R^2 = 0.058$  indicates that the debt–equity ratio and WCM explain 5.8% of the variance in ROE value. Our results are in line with the previous study by Berger and Di Patti [5] and Gill et al. [6], and are in favor of our stated hypothesis (H2). This indicates that the greater the proportion of funds financed by debt and the greater the WCM efficiency, the higher the firm’s performance (ROE). Column (4) illustrates the relationship between the GICS sectors used in our study and the performance of an Indian firm.

The results reveal a non-significant coefficient value for all industries besides the real estate industry, which positively affects the firm’s performance significantly. Communication services, consumer staples, financial institutions, industrials, and materials perform poorly. However, consumer discretionary, energy, health care, and information technology (IT) industries demonstrate positive firm performance between 2016 and 2021.

Table 2 provides additional estimates for the dependent variable ROA used in the analysis. Column 6 evaluates the relationship between the financial autonomy ratio and ROA for Indian firms. According to the previous research results by Yang and Wang [11], from a sample of the top 50 e-commerce retailers in China, the regression analysis results show that when the rate of financial autonomy increases

<sup>6</sup> <https://assets.kpmg/content/dam/kpmg/in/pdf/2020/04/aau-covid-19-financial-reporting-impact-going-concern-rbi-measures-chapter-2.pdf>

**Table 2** Data analysis using regression results for ordinary least square (OLS), fixed effect model (FEM), and random effect model (REM)

	ROE			ROA		
	(2)	(3)	(4)	(5)	(6)	(7)
	OLS	FEM	REM	OLS	FEM	REM
Constant	− 0.056		0.080	0.030**		0.034*
Debt-equity ratio	0.076***	0.049***	0.064***			
Financial autonomy rate				− 0.054***	− 0.003	− 0.029***
Net Working capital	1.312***	0.973***	1.171***	0.084***	0.075***	0.082***
Communication services	− 0.034		− 0.113	− 0.010		− 0.021
Consumer discretionary	0.027		0.024	− 0.004		− 0.009
Consumer staples	− 0.077		− 0.113	− 0.010		− 0.019
Energy	0.180		0.244	0.006		0.002
Financial	− 0.096		− 0.188	− 0.010		− 0.021
Health care	0.163		0.182	0.011		0.004
Industrials	− 0.187		− 0.207	− 0.010		− 0.018
Information technology (IT)	0.053		0.002	− 0.003		− 0.011
Materials	− 0.076		− 0.063	− 0.002		− 0.009
Real estate	0.786**		0.659	0.023		0.018
R <sup>2</sup>	0.169	0.0585	0.115	0.150	0.044	0.091
Observations	9058	9058	9058	9058	9058	9058
Hausman test ( $\chi^2$ statistics)			85.646***			123.960***
			FEM is optimal			FEM is optimal

Note ROE return on equity, ROA return on asset. \* significant at the 0.1 significance level, \*\* significant at the 0.05 significance level and \*\*\* significant at the 0.01 significance level

by one percent, financial performance (ROA) decreases by 0.004 points. Thus, we can state that the results support the hypothesis (H3).

In addition, the estimated relationship between net working capital and firm performance (ROA) is positive, with a coefficient value of 0.075. The R<sup>2</sup> value of 0.044 indicates that the independent variables explain 4.4% of the variance in ROA. Table 3 demonstrates the impact of alternative net working capital management proxies.

In Table 3, for example, the quick ratio is used to validate its impact on the firm performance of the sample firms. Column (3) assesses the debt-to-equity ratio's effect

**Table 3** Data analysis using regression results for ordinary least square (OLS), fixed effect model (FEM), and random effect model (REM)

	ROE			ROA		
	(2)	(3)	(4)	(5)	(6)	(7)
	OLS	FEM	REM	OLS	FEM	REM
Constant	- 0.443**		- 0.134	0.009		0.018
Debt-equity Ratio	0.076***	0.046***	0.061***			
Financial autonomy Rate				- 0.059***	- 0.004	- 0.032***
Quick ratio	0.468***	0.203***	0.313***	0.027***	0.0151***	0.021
Communication services	0.038		- 0.076	- 0.005		- 0.017
Consumer discretionary	0.149		0.110	0.003		- 0.003
Consumer staples	0.127		0.024	0.002		- 0.009
Energy	0.294		0.327	0.013		0.007
Financial	0.005		- 0.133	- 0.003		- 0.016
Health care	0.249		0.237	0.017		0.008
Industrials	- 0.013		- 0.073	0.001		- 0.008
Information technology (IT)	0.015		- 0.022	- 0.004		- 0.011
Materials	0.088		0.471	0.007		- 0.001
Real estate	0.747		0.568	0.020		0.012
R <sup>2</sup>	0.149	0.040	0.090	0.108	0.016	0.052
Observations	9058	9058	9058	9058	9058	9058
Hausman test ( $\chi^2$ statistics)			219.3***			217.05***
			FEM is optimal			FEM is optimal

Note ROE return on Equity, ROA return on Asset. \* significant at the 0.1 significance level, \*\* significant at the 0.05 significance level and \*\*\* significant at the 0.01 significance level

on firms' ROE with the same results as Table 3. Column (3) in Table 3 explains the influence of quick ratio as an additional variable on ROE. The results indicate a positive association between the quick ratio and the company's performance (ROE). According to the findings, if the quick ratio rises in line with a prior study by Borhan et al. [13] that found a positive correlation between the quick ratio and the performance of Indian manufacturing firms, the value of ROE will increase by 0.203. Thus, it supports our stated hypothesis (H4).

The results are negative in the communication services, financial, industrial, and information technology sectors. However, the consumer discretionary, consumer

staples, energy, health care, materials, and real estate sectors have a positive relationship.

Table 3 provides additional estimates for the dependent variable ROA used in the analysis. Column (6) evaluates the impact of the firms' financial autonomy on their ROA with the same results as Table 3. Column (6) identifies the relationship between ROA and an additional explanatory variable utilized in the study. Column (6) results indicate a positive and statistically significant relationship between quick ratio and firm performance (ROA) with a coefficient value of 0.015. Column (7) reveals the outcomes of firm performance across various industries.

## 6 Conclusion

Using ROA and ROE as proxy variables for firm performance, our findings indicate that appropriate liquidity management and debt financing improve financial execution. However, financial autonomy is negatively associated with ROA. According to the study, the market's overall net earnings decreased by 52.782% during COVID-19 (2019–20). During COVID-19, the performance of the communication services, consumer discretionary, financial, industrial, material, real estate, and IT sectors declined significantly. However, there has been an improvement in the performance of firms in the energy and healthcare industries.

Our findings can facilitate decision-makers, such as management and shareholders, to make the most effective management and financing decisions for their operations. This research paper has some analytical limitations, such as the possibility of organizational bifurcation based on organization size (small, medium, and large) and industry, which may be further subdivided into subsectors to derive industry-specific impact. In addition, the study included only a small number of financial ratios. Future researchers may examine in greater depth the effect of liquidity and turnover ratios on financial performance. Researchers might also anticipate comparison analyses throughout three unique stages, namely before the COVID-19 period (2018–19), the COVID-19 period (2019–20), and after the COVID-19 period, in order to thoroughly understand the influence of COVID-19 on company performance (2021–22).

## References

1. Cortez RM, Johnston WJ (2020) The Coronavirus crisis in B2B settings: crisis uniqueness and managerial implications based on social exchange theory. *Ind Mark Manag* 88:125–135
2. Abbas J, Al-Sulaiti K, Lorente DB, Shah SAR, Shahzad U (2022) Reset the industry redux through corporate social responsibility: the COVID-19 tourism impact on hospitality firms through business model innovation. In: Economic growth and environmental quality in a post-pandemic world. Routledge, pp 177–201

3. Alsamhi MH, Al-Ofairi FA, Farhan NH, Al-Ahdal WM, Siddiqui A (2022) Impact of Covid-19 on firms' performance: empirical evidence from India. *Cogent Bus Manag* 9(1):2044593
4. Shen H, Fu M, Pan H, Yu Z, Chen Y (2020) The impact of the Covid-19 pandemic on firm performance. *Emerg Mark Financ Trade* 56(10):2213–2230
5. Berger AN, Di Patti EB (2006) Capital structure and Firm performance: a new approach to testing agency theory and an application to the Banking industry. *J Bank Finance* 30(4):1065–1102
6. Gill A, Biger N, Mathur N (2011) The effect of capital structure on profitability: evidence from the United States. *Int J Manag* 28(4):3
7. Zeitun R, Tian GG (2014) Capital structure and corporate performance: evidence from Jordan. *Australas Acc Bus Fin J. Forthcoming*
8. Altaf N, Ahmad F (2019) Working capital financing, firm performance and financial constraints: empirical evidence from India. *Int J Managerial Fin* 15(4):464–477
9. Sharma AK, Kumar S (2011) Effect of working capital management on firm profitability: empirical evidence from India. *Glob Bus Rev* 12(1):159–173
10. Shrivastava A, Kumar N, Kumar P (2017) Bayesian analysis of working capital management on corporate profitability: evidence from India. *J Econ Stud* 44(4):568–584
11. Yang Z, Shi Y, Wang B (2015) Search engine marketing, financing ability and firm performance in E-commerce. *Procedia Comput Sci* 55:1106–1112
12. Lokwang JN, Gichure J, Oteki EB (2018) Effect of retained profits on performance of supermarkets in trans Nzoia County, Kenya. *Int J Recent Res Commer Econ Manag* 5(2):65–72
13. Borhan H, Mohamed RN, Azmi N (2014) The impact of financial ratios on the financial performance of a chemical company: the case of LyondellBasell industries. *World J Entrepreneurship Manag Sustain Develop* 10(2):154–160
14. Yameen M, Farhan NH, Tabash MI (2019) The impact of liquidity on firms' performance: empirical investigation from Indian pharmaceutical companies. *Acad J Interdisc Stud* 8(3):212–212
15. Wooldridge JM (2002) Econometric analysis of cross section and panel data. The MIT Press Cambridge, MA
16. Achim MV, Safta IL, Văidean VL, Mureşan GM, Borlea NS (2021) The Impact of Covid-19 on financial management: evidence from Romania. *Economic Research-Ekonomska Istraživanja* 35(1):1–26
17. Gujarati DN, Porter DC (2009) Panel data regression models. *Basic econometrics*
18. Khurana MK, Jhala S (2020) The effect of Covid-19 crisis on consumption pattern of consumers in relation to their income level. *Int J Innov Eng Res Manag* 7(1):54–64
19. Schmidt W, Raman A (2012) When supply-chain disruptions matter. Harvard Business School, Boston, MA

# An Enhanced BERT Model for Depression Detection on Social Media Posts



R. Nareshkumar and K. Nimala

**Abstract** Depression and other forms of mental illness are relatively common, and it has been shown that these conditions have an effect on an entity's physical health. Newly, artificial intelligence (AI) technologies have been created to aid mental health practitioners, such as psychiatrists and psychologists, in decision-making based on the historic data of patients (for example, medical records, behavioral data, social media use, etc.). These AI methods are intended to help mental health clinicians treat patients more effectively. One of the most recent generations of AI technologies, deep learning (DL), has exhibited greater performance in a wide variety of real-world applications spanning from computer vision to health care. When adopting bidirectional encoder representations from transformers (Enhanced BERT), the authors of the current research offer a new framework that may quickly and accurately identify postings that are connected to anxiety and depression. In addition, an intelligence distillation approach is a present method for transferring information from a large pretrained model BERT to a reduced model in instruction to expand the performance and accuracy of the smaller model. Researchers made use of word2vec and BERT in order to effectively analyze and identify symptoms of melancholy and anxiety based on our very own 40,000 data collecting infrastructure based on Twitter, the most widely used of the social media platforms. Using the Enhanced BERT methodology, our system achieves an accuracy of 92%, which is difficult than any other state-of-the-art technology.

**Keywords** Depression · Natural language · Mental illness detection · BERT

---

R. Nareshkumar · K. Nimala ()

Department of Networking and Communication, SRM Institute of Science and Technology,  
Kattankulathur, Chennai, India  
e-mail: [nimalak@srmist.edu.in](mailto:nimalak@srmist.edu.in)

R. Nareshkumar  
e-mail: [nr7061@srmist.edu.in](mailto:nr7061@srmist.edu.in)

## 1 Introduction

Today, there is a growing interest in a technique known as sentiment analysis, which aims to comprehend the feelings of individuals in a variety of settings pertaining to their everyday lives. Text data, emoticons, emojis, and other forms of social media expression, such as smiley faces and other symbols, are included in social media data, which would be used throughout the whole of the process, including the analysis and classification steps.

Processing of natural language is a mechanism that allows for communication with an intelligent system via the use of a natural example, such as English. It is capable of carrying out a wide variety of responsibilities on these sophisticated systems. The process of analyzing a sentence's lexicon entails locating and dissecting the words' unique constructions within the context of the phrase.

Sentiment analysis is the process of determining whether a body of text should be categorized as having a positive, negative, or neutral tone. The primary objective will be to examine the interests of individuals in such a manner that it will be beneficial to the growth of enterprises. It is a representation not just of polarity (positive, negative, and neutral), but also of feelings. A mental or physical condition that influences how a person thinks, feels, and behaves. Depression shows itself via a variety of moods, the most common of which is sorrow, as well as a lack of interest in things that you formerly found delightful. These symptoms may make it difficult to function, on the job work and at home.

We have carried out a thorough analysis with the purpose of assisting researchers in better comprehending the significance of emotion information in the diagnosis of mental diseases, as well as how the creation of emotion fusion techniques might support this diagnosis.

Our primary emphasis is on techniques for merging information about textual emotions to aid in the diagnosis of mental disorder.

In the next section, various depression-related work strategies that may assist in identifying mental illness are presented and classified. In Sect. 3 we delve into various methods and techniques, encompassing some of these topics. The fourth section discusses the numerous difficulties associated with depression emotions and analyzes outcomes. In the end, the work is wrapped up in Sect. 5, which discusses possible future approaches of research.

## 2 Related Work

The usage of social media has grown at a rapid pace over the course of the last decade, which has resulted in the creation of new chances to access and make sense of material provided by users. Accordingly, the detection of depression via the use of data from social media is making significant progress. Researchers have looked at

the prospect of diagnosing depressed symptoms or sadness based just on the content of what individuals post on social media platforms.

Shatte et al. [1] conducted research in an earlier study that investigated how machine learning methods may be used in the field of mental health. They examined the literature by classifying it. In a different piece of research, Durstewitz et al. [2] investigated a developing field of use for DL approaches in the field of psychiatry. They concentrated on DL in the research of brain dynamics and individuals' actions, and they provided the insights gained by integrating interpretable computational models into statistical context.

Clinical data, interviews, and surveys (including questionnaires [3] and rating scales [4]) are prominent types of data sources that may be used in the process of diagnosing mental disease from a clinical point of view. However, the production of them takes a significant amount of time and money. In addition, the majority of persons who have mental illnesses will not seek therapy [5], yet they will often communicate their emotions or moods via social media. The Linguistic Inquiry and Word Count (LIWC) [6] text analysis program counts words that correspond to psychological, linguistic, and thematic categories that show distinct emotional, social, and cognitive processes. The main part of the software does a comparison of each word in a given text to the list of terms found in the LIWC dictionary and then computes the percentage of words that fall into each of the many categories. LIWC has been employed in a great number of research because of how easy it is to assess the emotional states that are represented in the text.

Trotzek et al. [7] defined that linguistic information is significant in EDD text sequences, and identification utilized advantage of better model obtained from Wikipedia and trained using fastText and GloVe. Matero et al. [8] argued that the ability to diagnose sadness based on only one source of user writings is restricted. As a result, they suggested that three distinct kinds of traits or dimensions can be used to make predictions rather than simply textual data alone.

William et al. [9] systematic literature reviews (SLRs) are conducted to locate, evaluate, and analyze relevant literature in order to address specific research questions. The investigation explores inquiries about the detection of mental illness through text, revealing that early detection of depression on social media is feasible due to the presence of specific traits in the way individuals with mental disorders use their accounts on the platform.

Tadesse et al. [10] identify a vocabulary of words that are used more often by sad users. These findings demonstrate that our suggested approach may effectively enhance operational precision. In order to diagnose depression with 80% accuracy and 0.80 F1-scores, the most reliable single component is bigram using the support vector machine (SVM) classifier.

Dalal et al. [11] examined a wide range of machine learning and deep learning strategies on Twitter and Reddit datasets with variable user counts and post-densities. Various percentages of diagnosed users and control users are used to measure the effect of an unbalanced data collection. Results—according to the findings, SVM has correctly classified depressed users at a rate of 68% (70 in the diagnosis group

**Table 1** Comparison of previous approaches for depression detection

Author (year)	Source/illness type	Methods	Emotion feature extracted
[26] 2013	Twitter	SVM	Emotion
[24] 2015	Blog	LR	Emotion
[27] 2019	Twitter/mental disorder	Sentiment polarities' algorithm	Sentiment
[28] 2020	Depression	NB	Sentiment
[29] 2022	Twitter/suicide	LR, SVM, RF, XGBoost	Sentiment, emotion

and 70 in the control group). Performance drops for 150 people in each group, but recovers for 350 and 550 users.

Li et al. [12] HierBERT is a two-tiered hierarchical categorization approach that incorporates a BERT model with linguistic data and historical context. Utilizing a two-level hierarchical classification technique referenced, we integrate BERT and enhance the accuracy of geolocation predictions.

Singh [13] assessed the likelihood that the person is sad using extra factors, such as time, to the textual data already existing in the user's social media postings. The written work constructs the model with the use of predictive analytic techniques. The built-in classifier models identify the text's sentiment depending on the user's mood, which may be classified as depressed or non-depressive behavior.

Patidar et al. [14] depressive disorders affect 16 percent of young adults aged 10–19. Depression may have fatal outcomes in the worst cases. Users' productivity, academic performance, and social interaction suffer when depressed. Analyzing people's actions on social media might be a solution to this issue.

Gupt et al. [15] numerous indicators of depression's beginnings may be gleaned from a person's social media activity, including a lack of interest in others, a focus on one's own needs and interests, and increased day- and nighttime waking activity. For identifying signs of sadness in tweets, our team employed five different machine learning classifiers: decision trees, K-nearest neighbor, support vector machines, logistic regression, and long short-term memory (LSTM). Oversampling methods are analyzed by collecting the data in both a balanced and an unbalanced formats. A variety of strategies, including various techniques and feature extraction, were reviewed, and Table 1 displays their results.

### 3 Methodology

In this part of the article, research will discuss the structure that will be used to collect, analyze, and categorize data from social networking sites on mental health conditions such as anxiety and depression.

**Table 2** Classifier models on our built dataset incorporating our offered approach

Models	Accuracy	Recall	F1-score
SVM	0.51	0.56	0.46
RF	0.59	0.54	0.56
CNN + ALBERT	0.72	0.51	0.58
BiLSTM + ALBERT	0.69	0.54	0.61
BiGRU + ALBERT	0.68	0.53	0.6
BiLSTM + Enhanced BERT (Our Proposed Model)	<b>0.92</b>	<b>0.75</b>	<b>0.85</b>

The suggested overall design is shown in Fig. 1, which may be found in this body of research. Monitoring people's comments and postings on social media sites may give valuable insight into the ways in which individuals self-disclose and discuss concerns related to their mental health.

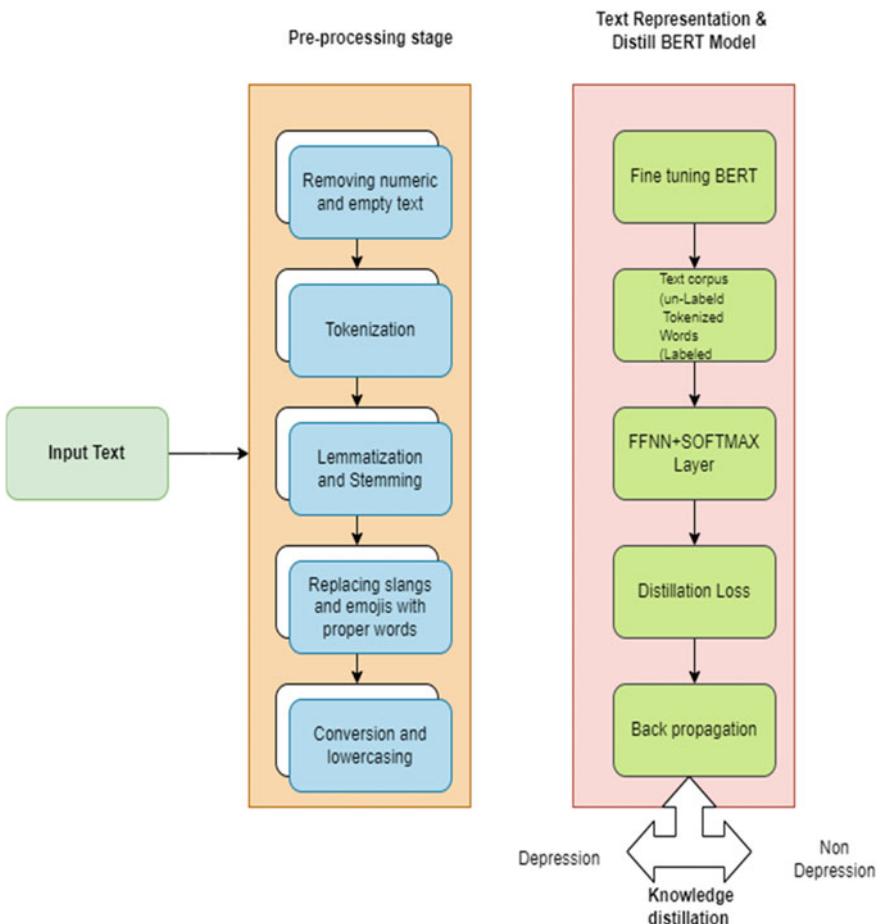
Thus, the hands-on structure that has been developed includes a variety of modules, such as labeling approaches, categorization, word embedding, and data gathering.

The use of fine-tuning and model optimization (Enhanced BERT), in order to construct a more efficient and accurate depression and anxiety detection model, is the crucial aspect of this process. In this context, Enhanced BERT refers to the process of constructing a condensed model by instructing it step-by-step in a sequential way using a more extensive pretrained BERT mode. Figure 2 provides more description of our proposed Enhanced BERT-based sentiment analysis model. This Fig. 2 demonstrates how pretrained weights from BERT are utilized to develop a better and lighter model for detecting sadness and anxiety in the downstream task. The research investigation has been conducted using a teacher–student architecture, which is an example of a universal framework for the transmission of information.

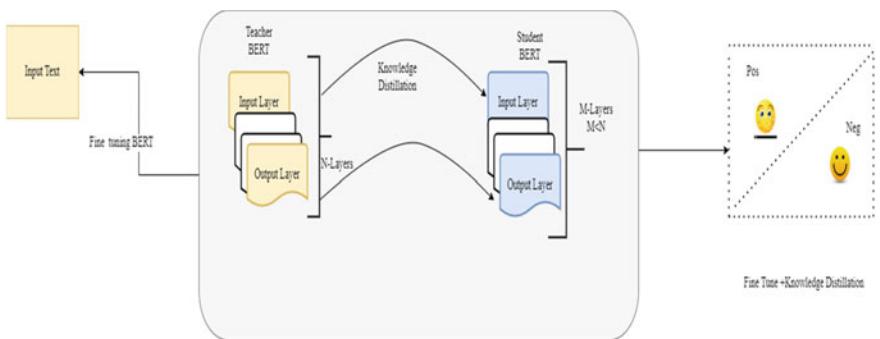
### 3.1 *Enhanced BERT Classical*

The complicated teacher network (BERT) is first accomplished on a excessive general dataset in order to prepare it to teach the student model. This stage calls for a high-performance computing environment, including high-performance graphics processing units (GPUs). Develop correspondence: While building a student network (a simplified version of BERT), there must be a link among the instant productions of the student model and those of the teacher modeling. Forward pass over the bigger typical: The information is transmitted over the instructor network to collect all of the rapid findings, and then information enhancement is applied.

Researchers used information based on responses, which often relates to the outcome of the last output layer of the instructor model (last logits). In recent years, response-based Enhanced BERT distillation has become more popular as a result of its reputation for being both easy and successful in the process of condensing huge



**Fig. 1** Our overall architecture for text representation of depression/anxiety detection uses fine-tuning, enhanced BERT distillation, and feature-based categorization



**Fig. 2** Enhanced BERT architecture for depression detection

models. In accordance with distillation, loss of response-based information may be expressed as follows:

$$L_D(p(Z_t), p(Z_s) = L_{KL}(p(Z_s), p(Z_t)), \quad (1)$$

where  $L_{KL}$  stands for the Kullback–Leibler divergence.

## 4 Results and Discussion

In the experiment that was carried out and investigated, our model qualities were examined alongside numerical performance enhanced BERT-based models and cutting-edge machine learning methods.

### 4.1 Dataset

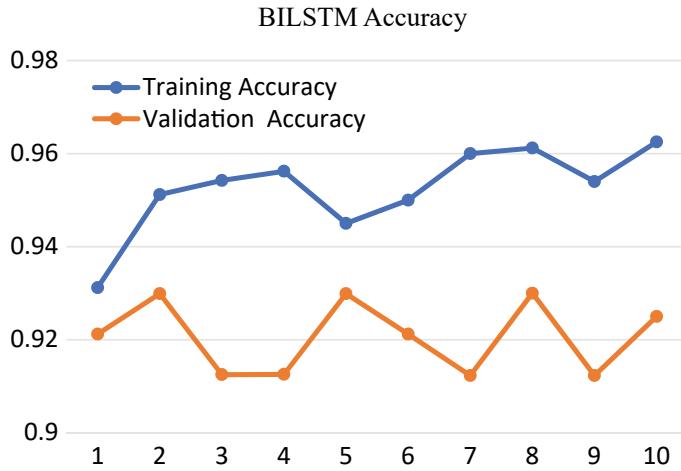
To achieve its objective, which is depression prediction, the inquiry that is being discussed here takes use of a conventional benchmark dataset that was established in the past. In this research, we look at how Twitter has adapted because of social media. As an example, we utilize a dataset comprising 40,000 social media postings collected from Twitter. We look into how people feel and what they are talking about. Using an unbalanced data collection, in which data from one class are disproportionately more numerous (or less) than data from other classes, is a typical source of error during data processing.

We gathered tweets and comments tagged with the term’s “depression” and “anxiety,” two often used subcategories [16], and categorized them as “positive,” while we gave the same classification to the normal tweets clean by keywords based on the circumplex prototypical.

The word2vec method takes in a collection of words and outputs a vector with the same information. Each of the words in the text will have to be assigned a vector in vector space, and the ensuing vector will have a large number of dimensions. The vectors are arranged in interplanetary such that words with comparable related meaning remain near together. Nevertheless, if a word is not encompassed in the training dataset, the word2vec method will not generate a vector representation for it.

After applying the model to the training data, you will obtain the training accuracy as part of the experiment’s conclusion; the test accuracy refers to the data used for the experiment itself. In our experiment, spanning 20 epochs, we designed a study to assess both model accuracy and model loss.

It is clear to observe that regardless of the various parameter, they continue to behave in the same manner. This little variation between our training and test accuracy demonstrates our ability to have a correct configuration of the regulation of all



**Fig. 3** Comparison of training and validation accuracies

constraints of the network as well as a data batch that is an accurate representation of the whole.

The appropriate hyperparameters and optimizer algorithms [17] for training our models are determined by the values of our past experiences. Batch size, epochs, the optimizer, and the amount of hidden layers were the inputs [18], and the results of our experiment demonstrated that Adam [19] was a good choice for the optimizer.

Hyperparameter tuning is implemented by carrying out a series of experiments, and then the results of these experiments are used to regulate the learning process.

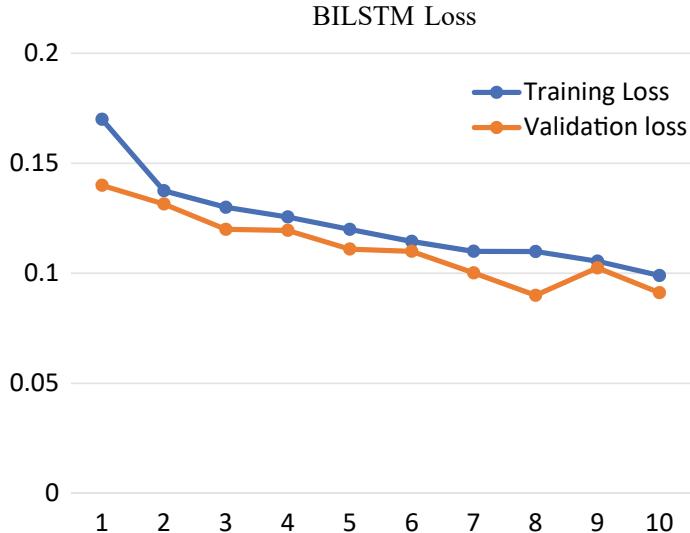
The outcomes of the tests performed using these parameter values, which are outlined in Table 2, produced the most accurate findings.

Figures 3 and 4 compare the effectiveness of our proposed Enhanced BERT [20] with BiLSTM in terms of both its accuracy and its loss throughout the training and validation phases.

Python is the programming language that is used in the process of actually carrying out the task that was suggested. Accuracy, recall, and the F1-score are the metrics [21] that are utilized in the process of measuring the efficacy of the models.

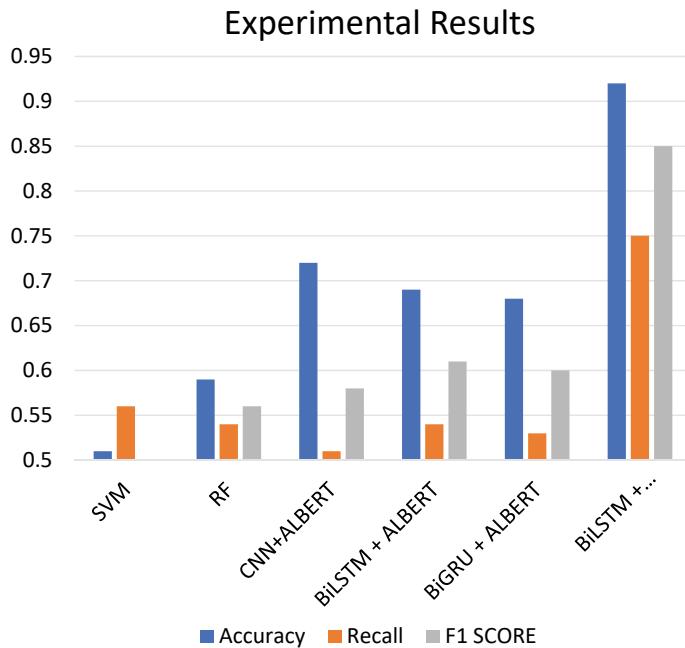
## 4.2 Comparison with the Existing System

Our comparison methodologies vary from those that have been suggested for depression detection to other methodologies for health-related text categorization projects [22]. We made use of our implementation of these baselines and made use of grid-search cross-validation to obtain the best values for each of the different approaches.



**Fig. 4** Comparison of training and validation losses

SVM [23]: The language vector was input into an SVM-based classifier, and the hinge loss was used as the objective function and acquired an good accuracy by using utilizing SVM and RF [24]. Researchers have shown that sequential approaches inside DL-based methods perform much better than CNN-based methods because they can capture better temporal context. This is because sequential methods are able to model stronger representations based on the context than CNN-based methods. Then utilized the language vectors generated using LMs and fed them to several RNN-based classifiers such as LSTM and GRU with cross-entropy as an objective function. These methods were described in [25]. As a baseline, we also made use of the TensorGCN + BiLSTM algorithm with cross-entropy as the goal function shown in Fig. 5.



**Fig. 5** Experimental results and comparing the performance of our model

## 5 Conclusion

Individuals' behaviors, thoughts, and moods are all impacted by their mental health when interacting with the external environment. In addition, issues relating to mental health are increasingly becoming a primary cause of disability, significantly contributing to the overall burden of illness. There are several challenges in the way of depression being detected and treated, some of which include a shortage of trained personnel in the health sector, societal stigma, and incorrect diagnoses. An experimental response-based implementation was carried out. Enhanced BERT accomplished an extraordinarily high level of accuracy in the identification of depressive and anxious states and fine-tuned the task. The approach suggested improves the correctness of smart healthcare classifications to identify mental health-related issues, including depression and anxiety. The findings that were obtained indicate that the suggested methodologies have the potential to assist in the creation of intelligent and effective structures for the diagnosis of sadness, anxiety, and other health-related issues based on the textual contents that users of social media platforms generate. In the future, effort may be done to construct a multimodal depression detection system that can make use of a wider variety of data types, such as text, picture, and behavioral aspects, to produce successful outcomes.

## References

1. Shatte ABR, Hutchinson DM, Teague SJ (2019) Machine learning in mental health: a scoping review of methods and applications. *Psychol Med* 49(09):1426–1448. <https://doi.org/10.1017/s0033291719000151>
2. Durstewitz D, Koppe G, Meyer-Lindenberg A (2019) Deep neural networks in psychiatry. *Mol Psychiatry* 24(11):1583–1598. <https://doi.org/10.1038/s41380-019-0365-9>
3. Kroenke K, Spitzer RL, Williams JBW (2003) The patient health questionnaire-2. *Med Care* 41(11):1284–1292. <https://doi.org/10.1097/01.mlr.0000093487.78664.3c>
4. von Glischinski M, Teismann T, Prinz S, Gebauer JE, Hirschfeld G (2016) Depressive symptom inventory suicidality subscale: optimal cut points for clinical and non-clinical samples. *Clin Psychol Psychother* 23(6):543–549. <https://doi.org/10.1002/cpp.2007>
5. Marcus M, Yasamy MT, van van Ommeren M, Chisholm D, Saxena S (2012) Depression: a global public health concern. *PsycEXTRA* dataset. <https://doi.org/10.1037/e517532013-004>
6. Meier T et al (2019) ‘LIWC auf Deutsch’: the development, psychometrics, and introduction of DE-LIWC2015. <https://doi.org/10.31234/osf.io/uq8zt>
7. Trotzek M, Koitka S, Friedrich CM (2020) Utilizing neural networks and linguistic metadata for early detection of depression indications in text sequences. *IEEE Trans Knowl Data Eng* 32(3):588–601. <https://doi.org/10.1109/tkde.2018.2885515>
8. Matero M et al (2019) Suicide risk assessment with multi-level dual-context language and BERT. In: Proceedings of the sixth workshop on computational linguistics and clinical psychology. <https://doi.org/10.18653/v1/w19-3005>
9. William D, Suhartono D (2021) Text-based depression detection on social media posts: a systematic literature review. *Procedia Comput Sci* 179:582–589. <https://doi.org/10.1016/j.procs.2021.01.043>
10. Tadesse MM, Lin H, Xu B, Yang L (2019) Detection of depression-related posts in reddit social media forum. *IEEE Access* 7:44883–44893. <https://doi.org/10.1109/access.2019.2909180>
11. Dalal S, Jain S, Dave M (2023) An Investigation of data requirements for the detection of depression from social media posts. *Recent Patents Eng* 17(3). <https://doi.org/10.2174/187221211662208121110956>
12. Li M, Lim KH (2022) Geotagging social media posts to landmarks using hierarchical BERT (Student Abstract). *Proc AAAI Conf Artif Intell* 36(11):12999–13000. <https://doi.org/10.1609/aaai.v36i11.21636>
13. Kumar Singh K (2023) Study of early risks of depression by analysing social media posts. *IIMS J Manag Sci* 14(1):9–25. <https://doi.org/10.1177/0976030x221112529>
14. Patidar H, Umre J (2021) Predicting depression level using social media posts. *Int J Res Granthaalayah* 8(12):234–237. <https://doi.org/10.29121/granthaalayah.v8.i12.2020.1972>
15. Gupta S, Goel L, Singh A, Prasad A, Ullah MA (2022) Psychological analysis for depression detection from social networking sites. *Comput Intell Neurosci* 2022:1–14. <https://doi.org/10.1155/2022/4395358>
16. Raja MS, Raj LA, Arun A (2022) Detection of depression among social media users with machine learning. *Webology* 19(1):250–257. <https://doi.org/10.14704/web.v19i1/web19019>
17. Nareshkumar R, Nimala K (2023) Interactive deep neural network for aspect-level sentiment analysis. In: 2023 International conference on artificial intelligence and knowledge discovery in concurrent engineering (ICECONF). <https://doi.org/10.1109/iceconf57129.2023.10083812>
18. Nareshkumar R, Agalya K, Arunpandiyar A, Vijayalakshmi M, Ranjani V, Ramya A (2023) An effective deep learning based recommender system with user and item embedding. In: 2023 International conference on artificial intelligence and knowledge discovery in concurrent engineering (ICECONF). <https://doi.org/10.1109/iceconf57129.2023.10083578>
19. Sailesh LJ, Kumar VK, Nimala K, Nareshkumar R (2023) Emotion detection in instagram social media platform. In: 2023 international conference on artificial intelligence and knowledge discovery in concurrent engineering (ICECONF). <https://doi.org/10.1109/iceconf57129.2023.10083724>

20. Nareshkumar R, Nimala K (2022) An exploration of intelligent deep learning models for fine grained aspect-based opinion mining. In: 2022 international conference on innovative computing, intelligent communication and smart electrical systems (ICSES). <https://doi.org/10.1109/icses55317.2022.9914094>
21. Sirenjeevi P, Karthick JM, Agalya K, Srikanth R, Elangovan T, Nareshkumar R (2023) Leaf disease identification using ResNet. In: 2023 international conference on artificial intelligence and knowledge discovery in concurrent engineering (ICECONF). <https://doi.org/10.1109/iceconf57129.2023.10083963>
22. Nareshkumar R, Suseela G, Nimala K, Niranjana G (2022) Feasibility and necessity of affective computing in emotion sensing of drivers for improved road safety. In: Advances in computational intelligence and robotics, pp 94–115. <https://doi.org/10.4018/978-1-6684-3843-5.ch007>
23. Nareshkumar R, Nimala K (2023) Interactive deep neural network for aspect-level sentiment analysis. In: 2023 international conference on artificial intelligence and knowledge discovery in concurrent engineering (ICECONF). Chennai, India, 2023, pp 1–8. <https://doi.org/10.1109/ICECONF57129.2023.10083812>
24. De Choudhury M, Gamon M, Counts S, Horvitz E (2021) Predicting depression via social media. Proc Int AAAI Conf Web Soc Media 7(1):128–137. <https://doi.org/10.1609/icwsm.v7i1.14432>
25. Reece AG, Reagan AJ, Lix KLM, Dodds PS, Danforth CM, Langer EJ (2017) Forecasting the onset and course of mental illness with Twitter data. Sci Rep 7(1). <https://doi.org/10.1038/s41598-017-12961-9>
26. Naseem U, Dunn AG, Kim J, Khushi M (2022) Early identification of depression severity levels on reddit using ordinal classification. In: Proceedings of the ACM web conference 2022. <https://doi.org/10.1145/3485447.3512128>
27. Ren F, Kang X, Quan C (2016) Examining accumulated emotional traits in suicide blogs with an emotion topic model. IEEE J Biomed Health Inform 20(5):1384–1396. <https://doi.org/10.1109/jbhi.2015.2459683>
28. Zhou T, Hu G, Wang L (2019) Psychological disorder identifying method based on emotion perception over social networks. Int J Environ Res Public Health 16(6):953. <https://doi.org/10.3390/ijerph16060953>
29. Razak CSA, Zulkarnain MA, Hamid SHA, Anuar NB, Jali MZ, Meon H (2020) Tweep: a system development to detect depression in Twitter posts. Comput Sci Technol 543–552. [https://doi.org/10.1007/978-981-15-0058-9\\_52](https://doi.org/10.1007/978-981-15-0058-9_52)

# Quality Prediction of a Stack Overflow Question Using Machine Learning



Tanvi Mehta, Samruddhi Multaikar, Srushti Patil, and Namrata Gawande

**Abstract** The support forums for developers are becoming more and more popular. Despite the fact that many programmers view adaptable knowledge as a precious asset, there could be problems with the accuracy of the information obtained. Morphological segmentation processes have progressed quickly as a consequence of the massive increase in digital resources. It may be difficult to distinguish between the quality of the questions and all of their substance on Q&A platforms. One such prevalent Q&A website that is important for programmers, developers, researchers, etc., to get technical information, especially in the field of Computer Science is Stack Overflow. Recently emerging machine learning optimization techniques, which make use of the most significant advancements in sophisticated learning methodologies, enable the autonomous retrieval of expressive characteristics. There are multiple methods for transforming human language into system-interpretable data as a result of the technological advancement in these approaches. This research presents various techniques that are implemented to predict the quality of questions. To analyze the textual data of 60,000 questions, the Lexicon-based Sentiment Analysis concept of Natural Language Processing (NLP) is utilized. Optimized methods like Naive Bayes (NB), Support Vector Classifier (SVC), K-Nearest Neighbors (KNNs), Decision Tree (DT), Random Forest (RF), and Logistic Regression (LR) are employed on the model. Finally, a relative and absolute analysis is performed with a major emphasis on accuracy and novel results from various approaches.

**Keywords** Question quality · Stack Overflow · Machine learning · Natural Language Processing · Q&A forum

---

T. Mehta (✉) · S. Multaikar · S. Patil · N. Gawande  
Pimpri Chinchwad College of Engineering, Pune, India  
e-mail: [tanvi.mehta19@pccoepune.org](mailto:tanvi.mehta19@pccoepune.org)

S. Multaikar  
e-mail: [samruddhi.multaikar19@pccoepune.org](mailto:samruddhi.multaikar19@pccoepune.org)

S. Patil  
e-mail: [srushti.patil19@pccoepune.org](mailto:srushti.patil19@pccoepune.org)

N. Gawande  
e-mail: [namrata.gawande@pccoepune.org](mailto:namrata.gawande@pccoepune.org)

## 1 Introduction

Due to the abundance of data accessible, a performance gap must often be addressed in order to accomplish a task [1]. One of the most fundamental methods of learning is raising queries. Historically, question-asking is used to be exclusive to schools. Yet it was required to be done outside of the teaching process because of a shortage of facilities [2].

A forum for discovering and learning more regarding a subject is offered by the Community-driven Question Answering (CQA) portal, Stack Overflow [3]. Because it offers a quick way to locate details on preferred themes, this site is quite prominent [4]. It presents insight into a vast range of subjects, from computing to philosophical inquiries.

Each internet individual has access to Stack Overflow's information, which includes both the concerns and the solutions [5]. For the forum to be used effectively, the caliber of the questions and answers must be attained. The caliber of the queries has an impact on how well CQA websites work [6]. Most frameworks view subjective questions as being unintentionally deceptive and not genuinely intended to increase knowledge [7]. Out-of-bounds inquiries, those that are inadequately phrased, and that are unsuitable may have an impact on the website's benchmarks or lead to the inquiry being removed.

The complexity of the question, wordiness, implications to the topic, phrases utilized, customer profile, question labels, the subjective experience of the question, simplification, restricted scope, etc., can all have an impact on the quality of the question [8]. Assessing and refining such attributes can assist the CQA service of Stack Overflow. This may aid in enhancing the question's consistency to produce satisfying responses and high ratings [9].

As a resource for knowledge sharing and finding, Stack Overflow is gradually becoming more well-liked. Users are drawn to Stack Overflow because it offers a simple and quick approach to locating the information they are looking for [10]. Understanding excellent inquiries helps enhance website offerings and customer satisfaction. In particular, the prediction of query quality and examination of the factors influencing it are carried out [11]. It would be possible to enhance the question formation, and subsequently the web content by having a better knowledge of something like the phrases used in both low- and high-quality inquiries.

This research intends to develop a web application that will be used to check and analyze the quality of the question before posting it on the website. Different NLP and ML strategies are implemented to achieve accurate results. The findings of the suggested technique, which uses a machine learning model based on the NB, DT, KNN, SVC, LR, and RF algorithms, demonstrate a comparison of these six methods. This will help to enhance the probability of getting a question answered which will indeed increase its readability. Moreover, it will also help Stack Overflow gain an advantage over rival question-and-answer websites.

Section 2 offers a review of the earlier academic articles. In Sect. 3, the procedures used in the aforementioned studies are fully described. The recommended work's

implementation is precisely described in Sect. 4. The findings of the study and observations are thoroughly examined in Sect. 5. The results and potential applications are addressed in Sect. 6.

## 2 Literature Review

Multiple papers and articles were examined and reviewed in light of the topic of interest. This research uses a variety of machine learning techniques to predict, from 2015 to 2022, the Q&A systems' respective quality scores.

In 2020, Li Xian Zhao et al. [12] emphasized that researching the problems may help to organize the advertising blitz, develop future advertising, and calculate expenditures. The authors suggested the VSAF method for examining changes in the amount viewed, changes in quantity replied, and changes in score immediately following the formation of the question using the fully CNN. The examination of the VSAF yielded a score of 90.97% accuracy using a random dataset.

In 2016, [13] Djedjiga Outioua et al. predicted Stack Overflow question ratings. According to the author, sourced information is a crucial resource for many engineers, and there may be rising concerns regarding the contents. The length of the program, an appropriate feedback score, the number of viewpoints, etc., are all determined here using multiple regression.

In 2022 [14], Garima Gupta et al. said that innovation had greatly improved. As a consequence, dynamically categorizing queries and using directed AI approaches may be used to discover this problem. They used SVM and LR to evaluate the outcomes depending on the Hamiltonian variables, and they were able to reach an 82.7% level of confidence.

In 2016, [15] Geoffrey Hodgins et al. provided research on the standards of queries with the intention of establishing the standards by which questions are judged and showing how to use this information to classify questions and replies with accuracy. Using several machine learning approaches, such as Random Forest, the prediction's accuracy was assessed to be 80%.

For the modeling process and quality control, Lakshmisri Surya et al. [16] proposed regularized statistical regression and network structures in 2019.

When the causes impacting the level of the questions were looked at, Baichuan Li et al. conducted a study in 2016 [17], and it was discovered that the interplay between the topic and the question asker produced different levels of question quality. The performance of the questions was predicted using the MRLP technique that they presented. MRLP, with a rate of 0.664, provided the highest level of accuracy.

Ming Li et al., in 2018 [18], proposed a reliable forecast for the solutions within a forum. The forecast had a  $P > 0.5$  probability, the researcher found after using Logistic Regression to measure the quality.

In 2022, Andrea Gasparetto et al. carried out research on the text classification technique [19]. Due to the rapid growth of techniques in NLP for textual to the

relevant information transformation format, there are currently an overwhelming number of methods to encrypt human speech into a computer format.

Using a variety of machine learning techniques, László Tóth et al. [20] reported the division of the inquiries into groups according to their syntactic and lexical aspects in a novel way in 2019. The method, which had a 74% accuracy rate, was predicated on the textual characteristics and used to determine whether the question should be closed or not.

### 3 Techniques

#### 3.1 Naïve Bayes (NB)

To perform classification tasks, this machine learning classifier that uses probabilities is employed. The Bayes theorem serves as the classifier’s foundation.

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

Each feature is isolated from the others and has an equal role in predicting the result [21]. When considering the value of the class variable, it “naively” assumes that every pair of features is conditionally not dependent. Due to its excellent scalability with a number of predictors, it is utilized to produce predictions in real-time.

#### 3.2 Support Vector Classifier (SVC)

The main objective of the Support Vector Classifier (SVC) with a linear kernel is to classify the data by finding an optimal hyperplane that best fits the data [22]. The class of new instances can then be predicted using the hyperplane based on their attributes. This property makes this method suitable for the task at hand.

$$w.x + b = 0$$

where,

$w$  = normal to hyperplane vector

$b$  = offset

### 3.3 Decision Tree (DT)

Decision Trees are part of the group of supervision training techniques utilized for solving issues with categorization and extrapolation. By learning decision-making processes, it seeks to determine the target variable [23]. The tree's root is where the class prediction procedure begins. Depending on the condition, trees split into their branches. The leaf node symbolizes the outcome. This makes it easy to explore relationships and understand features [24]. Decision Trees are more adaptable due to their ability to handle multiple outputs.

### 3.4 Logistic Regression (LR)

Modeling the link between the response of a categorical value and any number of predictive components is done using this analytical method [25]. It is applied to estimate the possibility of a binary outcome (such as success or failure) based on the values of the predictor variables. It is a popular algorithm in machine learning for classification tasks, such as determining whether an email is spam or not. The output of Logistic Regression is a probability score that indicates the likelihood of a certain outcome [26]. The function of the sigmoid is calculated by the equation that squashes return values between 0 and 1.

$$S(x) = \frac{1}{1 + e^{-x}}$$

### 3.5 K-Nearest Neighbors (KNN)

It works by identifying the K number of training examples that are closest to a new input data point and classifying it according to the predominant class among its closest neighbors [27]. By averaging the values of the K-nearest neighbors, KNN calculates the outcome of the incoming data point in the regression job. It is convenient to employ and suitable for low-dimensional datasets [28].

### 3.6 Random Forest (RF)

It is a powerful ensemble computing algorithm addressed for categorization and modeling problems [29]. During training, a large number of Decision Trees are constructed, and the mean (for regression) or mode (for classification) of each tree's

prediction is produced. A collection of characteristics is randomly chosen to create each Decision Tree and split the data based on the best feature to minimize impurity [30]. In contrast to single trees, Random Forest is a resilient method that can handle noisy, and high-dimensional datasets, and is also less prone to overfitting.

### ***3.7 Natural Language Processing (NLP)***

It implies to a machine's ability to understand both verbal and documented languages in a way that is comparable to a human. Raw text has to be encoded into a vector of numbers since it cannot be comprehended by machines. This makes use of the Bag-of-Words idea. It establishes the lexicon of well-known phrases and their existence helps in extracting features throughout the course of events by ignoring word order.

### ***3.8 Bag of Words (BoW)***

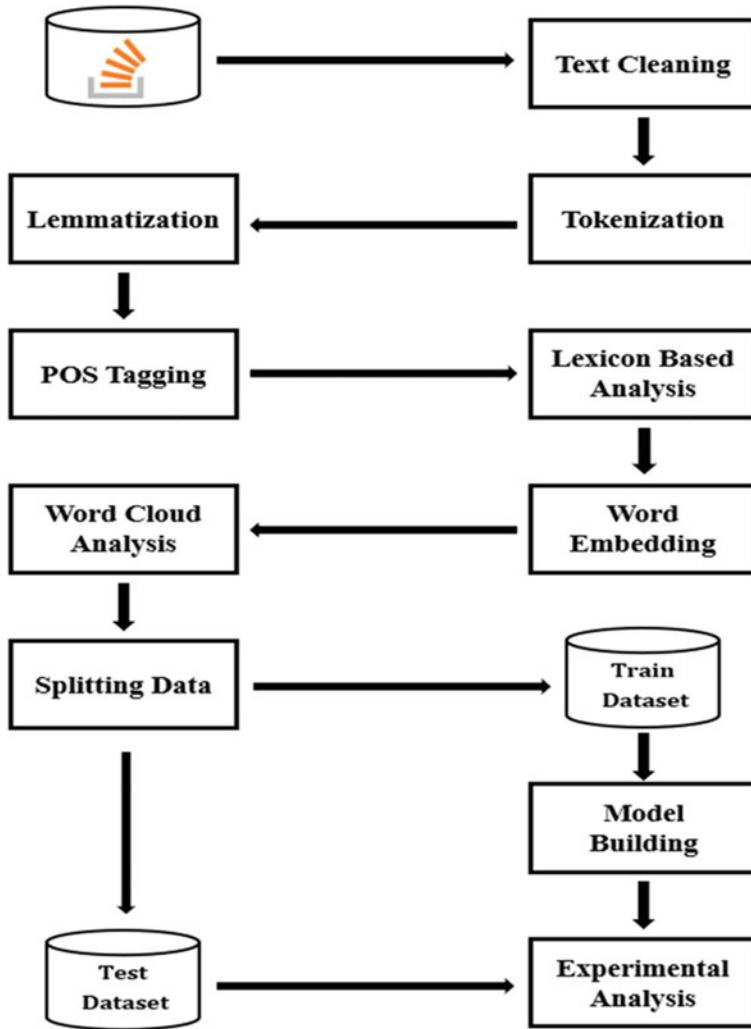
It is a way of describing corpora when text is being modeled using ML approaches. This approach transforms any corpus into firm-length vectors by analyzing the total number of instances every word arises. The vectorization process is the name given to this strategy. This simple approach addresses language modeling and literature categorization challenges by retrieving textual features. It provides several customization possibilities for data.

### ***3.9 Word Cloud Analysis***

Word clouds are visual illustrations of word harmonics that emphasize words that appear more often in the original content. The size of the word in the graphic represented the term's occurrence in the document. This type of visualization can assist evaluators with interpretive analysis of text by highlighting phrases that often appear in an accumulation of assessments, files, or other materials. It can also be used to emphasize the key points or themes during the reporting process.

## **4 Proposed Methodology and Implementation**

Before questions are posted on the Stack Overflow website, a consistent process for evaluating their quality is outlined in this area. Since the input is provided as text, NLP is used to conduct the data preparation procedure. The data are then sent to



**Fig. 1** Schematic representation of methodology

the model. Figure 1 provides an illustration of how the approach operates. A brief explanation of the flow is provided below.

#### 4.1 Importing Libraries

The libraries like numpy, pandas, matplotlib, etc., were imported. To perform the textual preprocessing, libraries such as sklearn, nltk, enism, and textblob are used.

## 4.2 Description of Dataset

The dataset acquires 60,000 questions which are retrieved from Stack Overflow. It consists of three types of questions: high quality, low-quality which can be editable, and low-quality questions that have to be discarded for better performance.

## 4.3 Cleaning Textual Data

Text cleaning in this context signifies taking measures to standardize the content, such as deleting irrelevant words or syllables, converting the text to subscripts. The dataset will be prepared to be utilized in subsequent processes after completing these tasks. It also includes the elimination of stop words using the enism package, dropna(), and drop duplicates commands.

## 4.4 Tokenization, Lemmatization, and PoS Tagging

Tokenization is breaking up phrases into individual words or tokens. These tokens are used to analyze the text deeper. It is done to make it simple to understand a tokens' meaning. Finally, for simplicity of observation, the morphemes are segmented into words. Moreover, it is followed by lemmatization and stemming to determine the semantics of the words used. Then, in computational semantics, grammatical tagging involves assigning a word in a corpus to a distinct component of speech relying on both its semantics and its relevance.

## 4.5 Lexicon-Based Sentiment Analysis

Using the lexical alignment of phrases that appear in a document, this phase involves computing sentiment. It was carried out with the help of the Python module TextBlob, which gives back subjectivity and polarity. Values for polarity are between  $[-1, 1]$ , with  $+1$  denoting a positive sentiment,  $0$  being neutral, and  $-1$  being negative. Subjectivity readings go between  $[0,1]$ , demonstrating whether the text is reflecting a possibility or an individual's view as in Fig. 2.



## 4.7 Splitting the Dataset

At this developmental stage, the applicable image is segmented into sets for testing and training. 80:20 is the ratio between training and testing.

## 4.8 Model Building and Performance Evaluation

The subsequent phase involves executing models like Naive Bayes, Supports Vector Classifier, Decision Tree, etc. These classifiers organize the training data in accordance with the training class labels that are provided for the classes.

## 5 Experimental Results and Analysis

Precision, recall, and F1-score are the outcomes attained from all the models. Positive and bad aspects of the book are separated. In Tables 1 and 2, the accurate findings are tabulated. Figure 4 shows a graph showing the accuracy of each model. It demonstrates that RF achieves a 97.56% accuracy rate.

The confusion matrix sheds light on how the models developed using machine learning performed on the testing data. Figures 5, 6, 7, 8, 9, and 10 display the confusion matrices for NB, SVC, DT, LR, KNN, and RF, respectively.

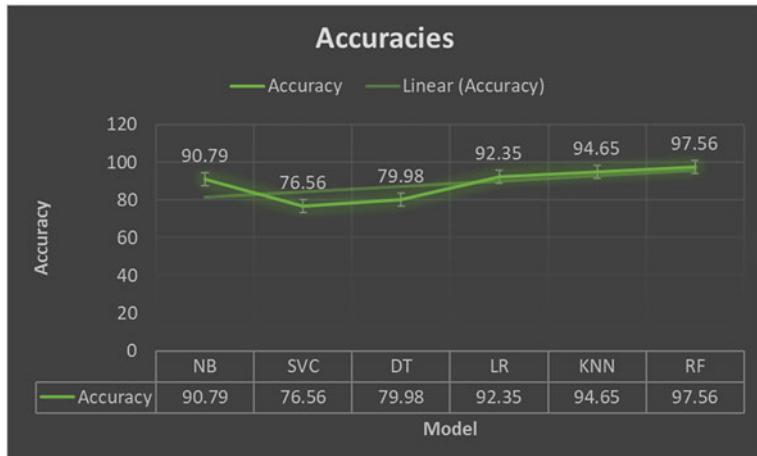
ROC curves are produced and used to assess the effectiveness of each categorization threshold. Figures 11 and 12 display the curves for DT and RF, respectively.

**Table 1** Comparative analysis of results

Models	Positive			Negative		
	Precision	Recall	F1-score	Precision	Recall	F1-score
NB	1.00	1.00	1.00	0.61	0.99	0.76
SVC	1.00	0.75	0.89	0.70	0.85	0.79
DT	0.77	0.90	0.62	1.00	0.65	0.59
LR	0.85	0.91	0.88	1.00	0.88	0.94
KNN	1.00	0.94	0.89	1.00	1.00	0.98
RF	1.00	1.00	1.00	1.00	1.00	1.00

**Table 2** Comparative analysis of accuracy

Models	NB	SVC	DT	LR	KNN	RF
Accuracy	90.79	76.56	79.98	92.35	94.65	97.56

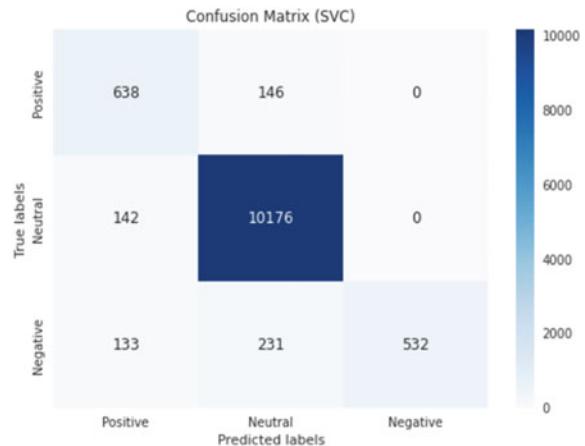


**Fig. 4** Accuracy of models

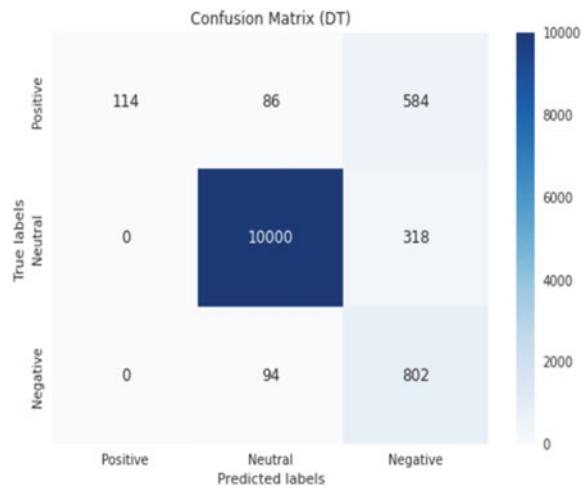
**Fig. 5** Confusion matrix of NB



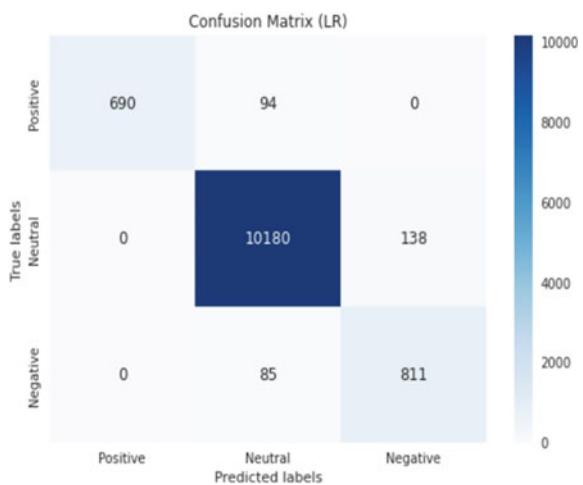
**Fig. 6** Confusion matrix of SVC



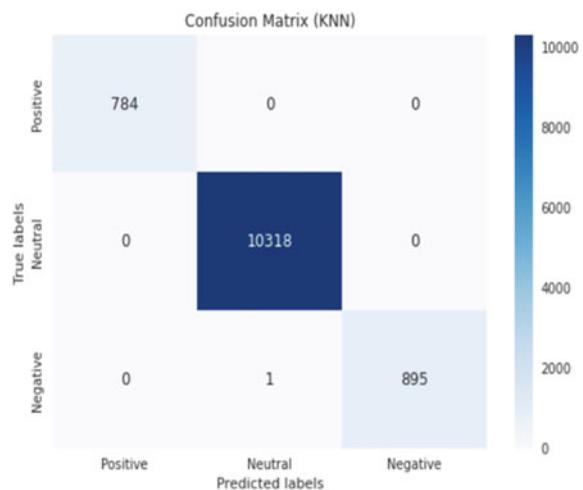
**Fig. 7** Confusion matrix of DT



**Fig. 8** Confusion matrix of LR



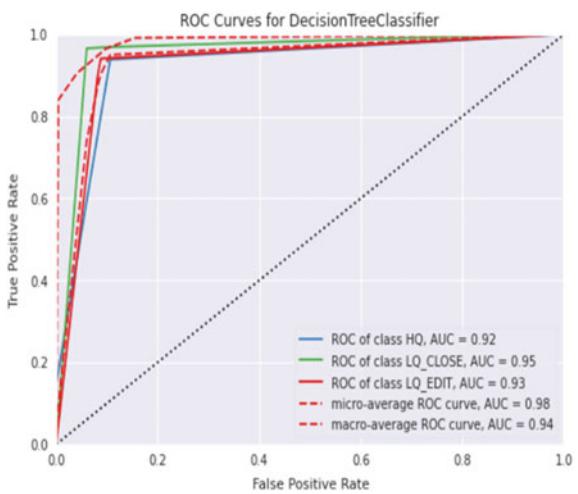
**Fig. 9** Confusion matrix of KNN

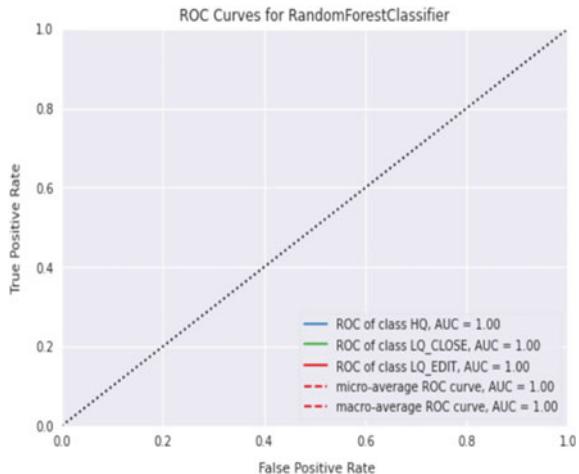


**Fig. 10** Confusion matrix of RF



**Fig. 11** ROC curves of DT



**Fig. 12** ROC curves of RF

## 6 Conclusion

This research employs multiple approaches and compares them to construct a model that analyzes the quality of the question. The strategies addressed in this paper can be decently implemented; however, the comparison is intended to determine the algorithm that best assesses the question's quality. Random Forest provides the highest accuracy of 97.56% of the algorithms mentioned. With the help of the above research, a web application that analyzes questions for quality and notifies the user of their high- or low-quality rating based on the question's semantics and phrasing was developed with the aid of the aforementioned research. Also, depending on the desired caliber of the work, a variety of correctional techniques, including phrase rephrasing, spelling and grammar correction, and the recommendation of a robust vocabulary, can be suggested. In order to enhance performance, deep learning models can also be deployed. Furthermore, the concept can be incorporated with cloud architecture by hosting data on cloud storage.

## References

1. Cha M et al (2020) Comparing and combining sentiment analysis methods. IEEE
2. Medhat W et al (2020) Sentiment analysis algorithms and applications: a survey. Elsevier
3. Agarwal A et al (2019) Sentiment analysis of twitter data. Association for Computational Linguistics
4. Chong WY et al (2019) Natural language processing for sentiment analysis. IEEE
5. Prabowo R, Thelwall M (2017) Sentiment analysis: a combined approach. Elsevier
6. Phan H et al (2018) Improving the performance of sentiment analysis of tweets containing fuzzy sentiment using the feature ensemble model. IEEE
7. Doaa Mohey et al (2019) A survey on sentiment analysis challenges. J King Saud Univ Eng Sci

8. Yadav A et al (2020) Sentiment analysis using deep learning architectures: a review. Springer
9. Rahmatika et al (2020) The effectiveness of Youtube as an online learning media. *J Educ Technol*
10. Novendri R et al (2020) Sentiment analysis of Youtube movie trailer comments using Naïve Bayes. *Bull Comput Sci Electr Eng*
11. Bozkurt AP et al (2020) Cleft Lip and palate Youtube videos: content usefulness and sentiment analysis. *Cleft Palate-Craniofacial J*
12. Outioua D et al (2020) Predicting questions' scores on stack overflow. In: International workshop on crowdsourcing in software engineering
13. Chrupala G et al (2016) Predicting the quality of questions on stack overflow. In: Proceedings of recent advances in natural language processing
14. Sharma D et al (2019) Tagging stack-overflow questions using supervised machine learning techniques. *Int J Eng Res Technol (IJERT)*
15. Zangari A et al (2016) A survey on text classification algorithms: from text to predictions. MDPI
16. Surya L (2022) Machine learning-future of quality assurance. *J Emerg Technol Innov Res (JETIR)*
17. Jiang J (2016) Hot question prediction in stack overflow. The Institution of Engineering and Technology
18. Hodgins G (2018) Classifying the quality of questions and answers from stack overflow. University of Dublin
19. Vidacs L et al (2019) Towards an accurate prediction of the question quality on stack overflow using a deep learning based-NLP approach. ResearchGate
20. Mehta T et al (2022) Intensification of agriculture using deep learning and machine learning. IEEE
21. Cambria E (2016) Affective computing and sentiment analysis. IEEE
22. Mehta T et al (2022) Analyzing Portfolio of biotech companies and predicting stock market using machine learning. IEEE
23. Dohaiha HH et al (2018) Deep learning for aspect-based sentiment analysis: a comparative review. Elsevier
24. Mehta T et al (2022) YouTube ad view sentiment analysis using deep learning and machine learning. *Int J Comput Appl* 184(11)
25. Das D et al (2018) Affective computing and sentiment analysis. Springer
26. Mehta T, Multaikar S, Patil S, Manolkar O, Gawande N (2022) A comparative study on approaches for text quality prediction using machine learning and natural language processing. IEEE
27. Gautam G et al (2014) Sentiment analysis of Twitter data using machine learning approaches and semantic analysis. In: Seventh international conference on contemporary computing (IC3)
28. Sahayak V et al (2020) Sentiment analysis on Twitter data. *Int J Innov Res Adv Eng*
29. Rudy et al (2009) Sentiment analysis: a combined approach. *J Informetrics*
30. Dang NC et al (2020) Sentiment analysis based on deep learning: a comparative study. MDPI

# Decision-Making Framework for Supplier Selection Using an Integrated Approach of Dempster–Shafer Theory and Maximum Entropy Principle



Garima Bisht and A. K. Pal

**Abstract** The process of supplier selection plays a crucial role in determining the success and competitiveness of organizations in today's dynamic business environment. To make informed decisions, decision-makers often rely on robust and efficient methods that consider multiple criteria simultaneously. In this regard, this paper presents a novel approach that combines the Dempster–Shafer theory and the Maximum Entropy Principle to address the supplier selection problem effectively. Dempster–Shafer theory, rooted in evidence theory, provides a powerful framework for handling uncertainty and incomplete information in decision-making. It allows decision-makers to represent and combine evidence from multiple sources, leading to more reliable and rational decisions. On the other hand, the Maximum Entropy Principle, a principle of statistical inference, is widely used for incorporating prior knowledge into decision-making processes. Our proposed approach harnesses the strengths of both the Dempster–Shafer theory and the Maximum Entropy Principle to enhance the supplier selection process. The decision-making framework is designed to handle the inherent uncertainty, subjectivity, and imprecision associated with the supplier selection problem. The flexibility and robustness of the method are demonstrated by the comparative and sensitivity analysis.

**Keywords** Dempster–Shafer theory · Cobb–Douglas utility function · Maximum entropy principle

---

G. Bisht (✉) · A. K. Pal

Department of Mathematics, Statistics and Computer Science, G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand 263145, India  
e-mail: [garimabisht98@gmail.com](mailto:garimabisht98@gmail.com)

A. K. Pal  
e-mail: [ak.pal@gbpuat-cbs.ac.in](mailto:ak.pal@gbpuat-cbs.ac.in)

## 1 Introduction

Supplier selection plays a critical role in the supply chain of organizations, and its quality depends on the effectiveness of the selection process [1]. Therefore, in-depth study of the supplier selection problem is not only of considerable theoretical value, but also has a high practical significance. The two main goals of supplier selection are to obtain optimum criteria weights and alternatives ranking. Over the past decades, several weight determination and ranking methods have been given by researchers [2–5]. Most of the traditional methods ignore the dependency among attributes. The attributes are not always independent of each other. Hence, extracting the dependency between attributes is of great implication for describing group decision-making processes in real-life scenarios.

MCDM approaches have garnered significant recognition and acceptance in effectively addressing real-world supplier selection problems [6]. Previous research [7–11] has made substantial contributions to the field by exploring diverse techniques for supplier selection. These techniques encompass methodologies such as the MULTIMOOSRAL method [12], the utilization of aggregation operators in a fuzzy environment [13], the integration of FAHP and FTOPSIS for steel manufacturing [14], FWASPAS for garment manufacturing [15], and MCDM-based approaches for aerospace and defense [16], oil and gas industries [17]. Wei and Zhou [18] employed an integrated approach combining BWM and VIKOR. Ayough et al. [19] solved the supplier selection problem using an integrated model consisting of base criterion and utility additive methods. These studies have contributed to the advancements of supplier selection methodologies.

The representation of uncertain information is a crucial aspect when it comes to supplier selection, as subjective judgments by human decision-makers introduce vagueness and ambiguity. To tackle uncertain information, this paper adopts the Dempster–Shafer evidence theory [20–23]. Dempster–Shafer’s theory of evidence has been a widespread concept among researchers for MCDM problems [24–27]. It is also helpful in situations when incomplete or contradictory information is provided by the decision-makers. The mechanism of D-S theory is based on collecting the evidence from different sources which are then effectively combined to reach a certain conclusion. However, it has not been used for determining the criteria weights.

In this study, we propose a novel weight determination method that considers the dependencies between attributes and incorporates Dempster–Shafer’s theory to handle uncertain and vague data encountered in decision-making problems. We employ the Cobb–Douglas utility function to calculate the loss value for each alternative. Furthermore, we apply the maximum entropy principle, which suggests that the probability distribution of the best alternative is the one with maximum entropy, to rank the alternatives based on their associated losses.

The rest of the paper is organized as follows—the basic concepts related to the study are discussed as preliminaries in Sect. 2. The different phases of the proposed methodology are presented in Sect. 3. An applied execution of the proposed approach with the help of an illustrative example of supplier selection is shown in Sect. 4.

Comparative analysis and experimental evaluations are discussed in Sect. 5, and sensitivity analysis is performed in Sect. 6, followed by conclusions in Sect. 7.

## 2 Preliminaries

### 2.1 Dempster–Shafer Evidence Theory

Dempster–Shafer theory also known as the theory of evidence or the theory of belief functions is a generalization of the Bayesian theory of subjective probability. It was proposed by Claussmann et al. [25] and later extended by [26]. It provides a mathematical framework for modeling uncertainty and a powerful method for combining the degree of evidence collected from different sources. Suppose  $\Theta = \{\theta_1, \theta_2, \dots, \theta_n\}$  referred to as the frame of discernment (FOD) is a finite set of mutually exclusive and exhaustive propositions. The set of the subsets of  $\Theta$ , i.e.,  $2^\Theta$  includes all the events and each subset or proposition will be assigned a value like probability based on evidence. The value 0 indicates no belief in the proposition; the value 1 indicates total belief in the proposition; and the value lying between 0 and 1 shows partial belief in the proposition. Hence, a basic probability assignment (BPA) or mass function ( $m$ ) over  $\Theta$  is described as a mapping  $m: 2^\Theta \rightarrow [0,1]$  satisfying the following condition:

$$m(\emptyset) = 0 \text{ and } \sum \sum_{A \in \Theta} m(A) = 1,$$

where  $A$  is one the propositions in  $2^\Theta$  and called focal element if  $m(A) > 1$ .

The belief function for  $A$  represented by  $\text{Bel}(A)$  measures the total belief committed to  $A$  by adding the mass of all proper subsets of  $A$ .

$$\text{Bel}(A) = \sum_{B \subset A} m(B). \quad (1)$$

And, the plausibility function  $pl(A)$  is defined as:

$$Pl(A) = \sum_{B \cap A = \emptyset} m(B) \quad (2)$$

The evidences collected from different sources are combined by using Dempster's rule of combination. Suppose  $m_1$  and  $m_2$  are the mass functions of two evidence  $A$  and  $B$  respectively, and Dempster's rule of combination is defined as:

$$m(C) = \frac{\sum_{A \cap B = C} N(A)N(B)}{\sum_{A \cap B = \emptyset} N(A)N(B)}. \quad (3)$$

## 2.2 Cobb–Douglas Utility Function

The Cobb–Douglas utility function is a well-known utility function that is used to represent the relationship between the amount of two or more inputs and the amount of output produced by these inputs. The general form of a Cobb–Douglas function over two alternatives is:

$$U(x_1, x_2) = x_1^a x_2^b. \quad (4)$$

The present work makes use of Cobb–Douglas function to determine the utility value of alternatives, let  $x_1, x_2 \dots x_n$  be  $n$  alternatives and  $w_1, w_2 \dots w_m$  be the weights of  $m$  criteria. Then, the Cobb–Douglas function can be written as:

$$U(x_1) = C_1(x_1)^{w_1} C_2(x_1)^{w_2} \dots C_m(x_1)^{w_m}. \quad (5)$$

## 2.3 Maximum Entropy Method

The ranking of alternatives is determined by maximum entropy method. The entropy is defined as:

$$S = \sum_{i=1}^n p(A_i) \ln\left(\frac{1}{p(A_i)}\right), \quad (6)$$

where  $p(A_i)$  is the probability distribution of alternatives used as score values to rank the alternatives. Since the sum of probability distribution is 1, we have  $\sum_{i=1}^n p(A_i) = 1$ .

The score values of alternatives can be computed by solving the following optimization problem:

$$\begin{aligned} \text{Max. } S &= \sum_{i=1}^n p(A_i) \ln\left(\frac{1}{p(A_i)}\right), \\ \text{s.t. } \sum_{i=1}^n p(A_i) &= 1, \\ G &= \sum_{i=1}^n p(A_i) g(A_i), \end{aligned}$$

where  $g(A_i)$  represents the expected loss of alternatives, and  $G$  represents the expected loss of positive solution.

### 3 Proposed Methodology

#### 3.1 Evaluate Criteria Weights

The steps involved in the evaluation of attribute weights are:

1. Find the dependency between the given sets of attributes using DEMATEL method.
2. Determine the evidence for attributes. The evidence for singletons is obtained by calculating the standard deviation of the normalized data, whereas the evidence of subsets of attributes is obtained by calculating correlation coefficients for the given combination of attributes as obtained in the first step.
3. The obtained evidences are used to categorize attributes into two frames of discernment, high and low based on Dempster–Shafer theory.
4. Rank the attributes using score function of high and low frames.
5. The weights of attributes are obtained by rank exponent weight method.

$$w_j = \frac{(n - r_j + 1)^p}{\sum_{k=1}^n (n - r_k + 1)^p},$$

where  $r_j$  represents the rank of  $j$ th criteria, and  $p$  represents the parameter describing the weights of criteria,  $j = 1, 2, 3 \dots m$ .

#### 3.2 Evaluate Alternative Ranking

The steps involved in the evaluation of attribute weights are:

1. Normalize the given decision matrix by using vector normalization.

$$n_{ij} = \frac{a_{ij}}{\sqrt{\sum_{i=1}^n (a_{ij})^2}}. \quad (7)$$

2. Find the positive ideal solution.

$$A^+ = \{ (n_{i1}, n_{i2}, \dots, n_{im}) | n_{ij} \text{ is the best value of } j \text{th attribute} \}.$$

3. Find the utility value of each alternative along with positive ideal solution.
4. Determine the loss of each alternative by calculating the difference between utility value of alternative and positive ideal solution.
5. Solve the optimization problem by using maximum entropy principle to get the score value of each alternative.
6. Rank the alternatives in descending order of score values.

**Table 1** Initial decision-making matrix

	C1	C2	C3	C4	C5
S1	2.5	1.67	6.67	7.5	1.67
S2	6.67	4.17	5.83	4.17	5
S3	5	8.33	7.5	5	5.83
S4	5.83	3.33	0.83	3.33	5.83

## 4 Numerical Example

For the validity of the proposed approach, we study a supplier selection MCDM problem.

### 4.1 Problem Description

Four suppliers are to be ranked for a refrigerator manufacturing industry based on the five criteria, quality, delivery time, price, technical capability, geographical location. The decision-making matrix is provided in Table 1.

### 4.2 Evaluation of Criteria Weights

The steps involved in finding criteria weights are:

1. The given decision matrix is shown in Table 1.
2. Normalize the decision matrix by using vector normalization as shown in Table 2.
3. The attributes are divided into cause-and-effect groups using DEMATEL. The dependency among attributes is defined as:

$$\begin{aligned} \{C_1\} &\rightarrow \{C_3\} \\ \{C_2\} &\rightarrow \{C_5\} \\ \{C_4\} &\rightarrow \{C_1, C_3\}. \end{aligned}$$

**Table 2** Normalized decision-making matrix

	C1	C2	C3	C4	C5
S1	0.238661	0.166455	0.573178	0.715983	0.170651
S2	0.636748	0.41564	0.500994	0.398087	0.510932
S3	0.477322	0.830283	0.644503	0.477322	0.595746
S4	0.556558	0.331914	0.071325	0.317897	0.595746

**Table 3** Attribute frame of discernment

	LI	HI
Singleton fuzzy measure	[0, 0.25]	[0.25, 0.5]
Subsets of fuzzy measure	[0.25, 0.75]	[0, 0.25] and [0.75, 1]

**Table 4** Final masses of DMs' evaluations

	LI	HI	$\theta$
C1	0.4171	0.0211	0.5617
C2	0.0000	0.8846	0.1153
C3	0.0550	0.8205	0.1243
C4	0.0342	0.8007	0.1649
C5	0.0392	0.8063	0.1544

**Table 5** Partition of attributes

High	C2 > C3 > C5 > C4
Low	C1

**Table 6** Attributes' weights

C1	C2	C3	C4	C5
0.1193	0.2667	0.2386	0.1687	0.2066

4. Determine the evidences of subsets of attributes.

$\{1\} = 0 \cdot 171$ ,  $\{2\} = 0 \cdot 282$ ,  $\{3\} = 0 \cdot 257$ ,  $\{4\} = 0 \cdot 171$ ,  $\{5\} = 0 \cdot 202$ ,  
 $\{1, 3\} = 0 \cdot 307$ ,  $\{2, 5\} = 0 \cdot 839$ ,  $\{1, 4\} = 0 \cdot 236$ ,  $\{3, 4\} = 0 \cdot 822$ .

The interval for the correlation coefficient is converted from  $[-1, 1]$  to  $[0, 1]$ .

5. The description of attributes frame of discernment as suggested by DMs is shown in Table 3. The (HI) and (LI) denote the attributes of high and low importances.
6. Final masses of decision-makers' evaluations are shown in Table 4.
7. Based on the final masses, the attributes are classified into high and low with the following preferences as shown in Table 5.
8. The ranking of attributes as obtained is: C2 > C3 > C5 > C4 > C1.
9. The attributes weights are determined by rank exponent weight method. The attributes weights are shown in Table 6.

### 4.3 Ranking of Alternatives

The steps involved in ranking alternatives are:

**Table 7** Loss associated with different alternatives

	Loss
S1	0.0948
S2	0.2685
S3	0.4170
S4	0.0688

**Table 8** Score values of different alternatives

	Score values	Rank
S1	0.260	2
S2	0.245	3
S3	0.232	4
S4	0.263	1

1. The normalized decision matrix is shown in Table 2.
2. Determine the positive ideal solution and utility value of all alternatives along with positive ideal solution.
3. The loss associated with different alternatives is shown in Table 7.
4. Solve the optimization problem with maximum entropy principle to get the score values of alternatives.
5. Finally, rank the alternatives in descending order of score values as shown in Table 8.

## 5 Comparative Analysis and Discussions

### 5.1 Comparative Analysis of Weight Determination Method

This section compares the weights of criteria as obtained by different weight determination methods. For this, we take the example as defined in Sect. 4. The study compares criteria weights as determined by entropy, standard deviation, and proposed method. Table 9 shows the criteria weights determined by each method and the related Pearson correlation coefficients ( $r$ ). Since  $r > 0.7$ , it establishes a strong similarity between the weights obtained by the proposed and other weight determination methods. The high correlation coefficients provide evidence of consistency and agreement among the different methods in determining the relative importance of the criteria. These results lend credibility and reliability to the attribute weights generated by the proposed method. Decision-makers can have confidence in utilizing these weights for effective and reliable decision-making in various problem contexts.

**Table 9** Weights and correlation coefficient of comparative analysis

Weights	Entropy	Standard deviation	Proposed method
w1	0.10717	0.15826	0.1193
w2	0.29889	0.26003	0.2667
w3	0.33405	0.23709	0.2386
w4	0.09194	0.15826	0.1687
w5	0.16793	0.18633	0.2066
<b>r</b>	<b>0.872248</b>	<b>0.928085</b>	

## 5.2 Comparative Analysis of Ranking Method

This section compares the proposed methodology with the existing MCDM methods based on the numerical examples. The problem of supplier selection presented in Sect. 4 is compared with the other well-known MCDM methods. The results are shown in Table 10.

The results from Table 10 are not completely identical with each other, but the best is preserved in each method; hence, the proposed method is stable to the optimal solution and can effectively be applied to solve decision-making problems.

Several shortcomings of previously developed approaches for supplier selection problems are discussed here. Kao et al. [15] integrated FAHP and FWASPAS for supplier selection, but FAHP can lead to inconsistencies in judgment and ranking criteria, making it difficult to achieve consensus among decision-makers due to the large number of pairwise comparisons. Additionally, FWASPAS may not be logical in certain circumstances. Gidiagba et al. [17] integrated TOPSIS and BWM for supplier selection, but the lack of consistency thresholds in BWM poses an important issue in determining the reliability of the results. Traditional MCDM approaches, though simple, have their limitations. TOPSIS merely sums up distances without considering their relative importance, PROMETHEE II suffers from high complexity due to a large number of iterations, and the normalization process of VIKOR is influenced by both ideal and non-ideal attributes values, thus potentially compromising the accuracy of results.

**Table 10** Comparison of ranking with different methods

The methods	Ranking results
Proposed method	S4 > S1 > S2 > S3
Method in [15]	S4 > S2 > S1 > S3
Method in [17]	S4 > S1 > S3 > S2
VIKOR	S4 > S2 > S1 > S3
TOPSIS	S4 > S1 > S2 > S3
PROMETHEE II	S4 > S2 > S1 > S3
WASPAS	S4 > S1 > S2 > S3
SAW	S4 > S1 > S2 > S3

The proposed method aims to address the limitations of existing supplier selection methods by incorporating attribute dependencies, handling uncertainty and vagueness, enhancing flexibility, and considering decision-makers' preferences.

1. Unlike traditional methods that assume attribute independence, the proposed method explicitly considers the interdependencies among attributes. By incorporating the functional dependencies between attributes, the method captures the relationships and interactions among criteria, resulting in more accurate evaluations and better-informed supplier selections.
2. The incorporation of Dempster–Shafer's theory in the proposed method allows for the handling of uncertain and vague information. By aggregating evidence, the method can effectively manage contradictory data provided by decision-makers, providing a more robust framework for decision-making under uncertainty.
3. The proposed method recognizes the significance of both quantitative and qualitative criteria in supplier selection. By employing the Cobb–Douglas utility function, it calculates the loss value for each alternative, taking into account both objective and subjective factors. This approach ensures a balanced evaluation that considers diverse criteria, including those related to supplier responsiveness, flexibility, and cultural fit.
4. The proposed method offers flexibility by allowing customization to different decision contexts. It does not rely on pre-defined weight determination and ranking techniques, enabling adaptation to specific industry requirements or organizational preferences. This flexibility ensures that the method can be tailored to the unique needs and characteristics of various supplier selection scenarios.
5. The proposed method acknowledges the importance of decision-makers' preferences in supplier selection. By incorporating Dempster–Shafer's theory and the maximum entropy principle, it can capture and integrate the diverse perspectives. This consideration ensures that the decision outcomes align with the collective interests and preferences of all decision-makers, leading to more inclusive and satisfactory supplier selections.

By addressing these limitations, the proposed method offers a comprehensive framework for supplier selection. It provides a more accurate, flexible, and stakeholder-centric approach, enabling decision-makers to make more informed and reliable supplier selection decisions in real-world scenarios.

## 6 Sensitivity Analysis

The aim of this subsection is to study the effect of most influential attribute on ranking results by proposed approach. For this, we generate four different scenarios by varying the weight of the most influential attribute and thus observing the change in the ranking results. Considering the original set of attribute weights,  $C_2$  attribute has been identified as the most influential. Table 11 shows the seven different scenarios by varying attribute weights.

**Table 11** Different scenarios for attribute weights

	Original	Set 1	Set 2	Set 3	Set 4
C1	0.1193	0.1609	0.1359	0.1109	0.0859
C2	0.2667	0.1	0.2	0.3	0.4
C3	0.2386	0.2802	0.2552	0.2302	0.2052
C4	0.1687	0.2104	0.1854	0.1604	0.1354
C5	0.2066	0.2483	0.2233	0.1983	0.1733

**Table 12** Ranking results with different weight vectors

Scenarios	Ranking results	Optimal alternative
Original	S4 > S1 > S2 > S3	S4
Set 1	S4 > S1 > S2 > S3	S4
Set 2	S4 > S1 > S2 > S3	S4
Set 3	S4 > S2 > S1 > S3	S4
Set 4	S4 > S2 > S1 > S3	S4

The ranking results obtained in different scenarios are shown in Table 12. It can be easily observed from Table 12 and Fig. 1 that although the attributes' weights differ greatly, a very small change in ranking results is seen. Hence, the proposed approach is stable in terms of ranking and under the condition of varying attribute weights. These findings affirm the robustness and reliability of the proposed approach, demonstrating its ability to maintain consistency in identifying the optimal project across different attribute weight configurations. Regardless of fluctuations in attribute weights, the proposed approach remains steadfast in delivering reliable and stable results.

**Fig. 1** Ranking results in different scenarios

—●— S1   —●— S2   —●— S3   —●— S4



## 7 Conclusions

The present study introduces a novel approach in the field of multi-criteria decision-making with the aim of enhancing the credibility of the obtained results. The main motivation behind this new approach is to address the interdependence among attributes, thereby improving the accuracy of weight determination and ranking. This is a crucial step toward achieving more accurate and informed decision-making outcomes. To effectively capture and handle uncertainty in the decision-making process, the well-established Dempster–Shafer (D-S) theory is employed. By leveraging the power of D-S theory, the proposed methodology provides a robust framework that accounts for uncertainties, leading to more reliable and rational decisions. The proposed approach involves the development of an optimization model that determines attribute weights. This model utilizes the Cobb–Douglas function to calculate the loss value associated with each alternative. By employing this function, the proposed methodology accurately evaluates the performance of each alternative, enabling decision-makers to make informed choices based on comprehensive evaluations. Furthermore, the maximum entropy principle, widely recognized in statistical inference, is applied to rank the alternatives based on their respective losses. This principle ensures that the probability distribution of the best alternative is maximized, resulting in more effective and meaningful rankings. To validate the practicality and efficacy of the proposed approach, a case study on supplier selection is conducted. The robustness and reliability of the proposed approach are rigorously evaluated through comparative and sensitivity analyses. The results of these analyses demonstrate the superiority and practicality of the proposed approach in achieving accurate and reliable decision-making outcomes.

## References

1. Karsak EE, Dursun M (2015) An integrated fuzzy MCDM approach for supplier evaluation and selection. *Comput Ind Eng* 82:82–93
2. Zardari NH, Ahmed K, Shirazi SM, Yusop ZB (2015) Weighting methods and their effects on multi-criteria decision-making model outcomes in water resources management. Springer, New York, NY, USA
3. Delice EK, Can GF (2020) A new approach for ergonomic risk assessment integrating KEMIRA, best-worst and MCDM methods. *Soft Comput* 24:15093–15110
4. Du YW, Gao K (2020) Ecological security evaluation of marine ranching with AHP-entropy-based TOPSIS: A case study of Yantai China. *Mar Policy* 122:104223
5. Bisht G, Pal AK (2023) Principal component analysis and correlation coefficient-based decision-making approach for stock portfolio selection. In: Tiwari R, Pavone MF, Saraswat M (eds) Proceedings of international conference on computational intelligence. ICCI 2022. Algorithms for intelligent systems. Springer, Singapore, pp 25–37
6. Vasiljević M, Fazlollahtabar H, Stević T, Vesković S (2018) A rough multicriteria approach for evaluation of the supplier criteria in automotive industry. *Decis Making Appl Manag Eng* 1(1):82–96
7. Alkan O, Albayrak OK (2020) Ranking of renewable energy sources for regions in Turkey by fuzzy entropy based fuzzy COPRAS and fuzzy MULTIMOORA. *Renew Energy* 162:712–726

8. Zavadskas EK, Bausys R, Lescauskiene I, Usovaite A (2021) MULTIMOORA under interval-valued neutrosophic sets as the basis for the quantitative heuristic evaluation methodology HEBIN. *Mathematics* 9(1):66
9. Behera DK, Beura S (2023) Supplier selection for an industry using MCDM techniques. *Mater Today Proc* 74:901–909
10. Singh A, Kumar V, Verma P (2023) Sustainable supplier selection in a construction company: a new MCDM method based on dominance-based rough set analysis. *Constr Innov*
11. Chai N, Zhou W, Jiang Z (2023) Sustainable supplier selection using an intuitionistic and interval-valued fuzzy MCDM approach based on cumulative prospect theory. *Inf Sci* 626:710–737
12. Ulutas A, Stanujkic D, Karabasevic D, Popovic G, Zavadskas EK, Smarandache F, Brauers WKM (2021) Developing a novel integrated MCDM MULTIMOOSRAL approach for supplier selection. *Informatica* 32:145–161
13. Bisht G (2023) A novel multi-criteria group decision-making approach using aggregation operators and weight determination method for supplier selection problem in hesitant Pythagorean fuzzy environment. *Decis Sci Lett* 12(3):525–550
14. Wang CN, Nguyen TL, Dang TT (2022) Two-Stage fuzzy MCDM for green supplier selection in steel industry. *Intell Autom Soft Comput* 33(2):1245–1260
15. Kao JC, Wang CN, Nguyen VT, Husain ST (2022) A fuzzy MCDM model of supplier selection in supply chain management. *Intell Autom Soft Comput* 31(3):1451–1466
16. Rasmussen A, Sobic H, Saha S, Nielsen IE (2023) Supplier selection for aerospace & defense industry through MCDM methods. *Cleaner Eng Technol* 12:100590
17. Gidiagba J, Tartibu L, Okwu M (2023) Sustainable supplier selection in the oil and gas industry: an integrated multi-criteria decision-making approach. *Procedia Comput Sci* 217:1243–1255
18. Wei Q, Zhou C (2023) A multi-criteria decision-making framework for electric vehicle supplier selection of government agencies and public bodies in China. *Environ Sci Pollut Res* 30:10540–10559
19. Ayough A, Shargh SB, Khorshidvand B (2023) A new integrated approach based on base-criterion and utility additive methods and its application to supplier selection problem. *Expert Syst Appl* 221:119740
20. Dempster AP (1967) Upper and lower probabilities induced by a multivalued mapping. *Ann Math Stat* 38(2):325–339
21. Shafer G (1976) A mathematical theory of evidence. Princeton University Press, Princeton
22. Mo H, Deng Y (2016) A new aggregating operator in linguistic decision making based on d numbers. *Int J Uncertain Fuzziness Knowl Based Syst* 24(6):831–846
23. Kang B, Chhipi-Shrestha G, Deng Y, Mori J, Hewage K, Sadiq R (2018) Development of a predictive model for clostridium difficile infection incidence in hospitals using Gaussian mixture model and Dempster-Shafer theory. *Stoch Environ Res Risk Assess* 32:1743–1758
24. Beynon M, Cosker D, Marshall D (2001) An expert system for multi-criteria decision-making using Dempster Shafer theory. *Expert Syst Appl* 20:357–367
25. Claussmann L, O'Brien M, Glaser S, Najjaran H, Gruyer D (2018) Multi-criteria decision making for autonomous vehicles using fuzzy Dempster-Shafer reasoning. *IEEE Intell Veh Symp Proc* 2195–2202
26. Silva LGDO, De Almeida-Filho AT (2016) A multicriteria approach for analysis of conflicts in evidence theory. *Inf Sci (Ny)* 346–347:275–285
27. Dutta P (2015) Uncertainty modeling in risk assessment based on Dempster-Shafer theory of evidence with generalized fuzzy focal elements. *Fuzzy Inf Eng* 7:15–30

# Improved Accuracy of Robotic Arm Using Virtual Environment



Utkarsh Rastogi, Javed Sayyad, B. T. Ramesh, and Arunkumar Bongale

**Abstract** Robotic arms are utilized in various fields, but their precision remains an issue, particularly in high-precision applications. Simulation and testing of the robotic arm's movements in a virtual environment can improve its accuracy. This paper examines how VE technology improves robot arm precision and summarizes relevant studies. A strategy for improving robotic arm accuracy using VE technology is proposed, along with simulation results. This work presents an improved method for improving the precision of a six-DOF robotic arm by utilizing MATLAB Simulink's VE, Fusion 360 for design, and SIMSCAPE for validation. This approach improves existing methods to enhance both performance and accuracy and guarantees precise control and movement confirmation.

**Keywords** Automation · Robotic arm · Virtual environment · MATLAB · Degree of freedom · SIMULINK · SIMSCAPE

## 1 Introduction

Envision an active generation floor where robotic arms tirelessly and accurately carry out complex occupations. These mechanical ponders have changed various segments, from healthcare to industry, by encouraging expanded efficiency, precision, and security [1]. Engineers and analysts, be that as it may, are continuously working to move forward the accuracy of these mechanical arms to unleash more potential. In this investigative article, we embarked on a journey to extend the accuracy of mechanical arms [2]. Our objective is to approach that not as it guarantees exact developments but too right away affirm the arm's rightness [3]. We need to reconsider what is conceivable regarding mechanical arm execution by using the control of virtual situations and advanced reenactment apparatuses [4]. As we dove more profound into this world of robotic arms, we experienced different challenges. Conventional strategies regularly

---

U. Rastogi · J. Sayyad (✉) · B. T. Ramesh · A. Bongale

Department of Robotics and Automation Engineering, Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Lavale, Pune, Maharashtra 412115, India  
e-mail: [jksayyad23@gmail.com](mailto:jksayyad23@gmail.com)

center, as it were, on optimizing arm directions without ultimately affirming their exactness. While orbital optimization is undoubtedly required, it does not capture the arm's genuine exactness. We recognize the need for an all-encompassing approach that combines plan, reenactment, and confirmation into one consistent workflow [5].

To set the organization, we investigate past inquiries about endeavors that cleared the way for our inventive approach. Jaemin et al., who bravely plunged into machine learning building to optimize the direction of a mechanical arm. Whereas their work guarantees to make strides in productivity and smoothness, it still clears out a genuine crevice in precision confirmation. Their accomplishments propelled them, but we saw they had to be assisted. We experienced the groundbreaking work of Jones et al., who introduced a vision-based framework for observing and following mechanical arm developments [6]. Their approach tackled the control of visual criticism to make strides in exactness. Be that as it may, we realized that exactness alone was insufficient. We must endeavour for exact control and real-time verification to guarantee the arms development is steady with the required result. Navigating the inquire-about scene, we experience pioneers such as Zhu et al., who have gone to extraordinary lengths to get the kinematic demonstration of the mechanical arm [7]. Their work has given good knowledge into theoretical perspectives of arms development, explaining the complex connections that oversee arm development. In any case, we see an opportunity to bridge the hole between hypothesis and hone, to turn this information into reasonable confirmation and observing procedures.

Drawing from the lessons learned, we set out for progression. Our strategy is planning the arrange capabilities of Fusion 360, a viable 3D modeling instrument, with the entertainment capacity of SIMSCAPE in MATLAB Simulink [8]. By combining these gadgets, we make a virtual environment that becomes our investigative office, play area, and testing ground. Interior this virtual space, characterize the specified headings for our mechanical arm, organizing its improvements with numerical precision. We thoroughly build a simulation system to examine each movement, comparing the arm's simulated movements to the desired directions. It is interior this confusing move of plan, diversion, and affirmation that we open the potential for unprecedented accuracy. Our paper is the epitome of this travel, confirming the control of improvement and collaboration [9]. Through our inquiry almost, we search to reconsider the limits of mechanical arm exactness, bringing one step closer to realizing the overall potential of these pivotal machines. As we meander into virtual circumstances, imaginative capacity becomes a reality, precision knows no bounds [10].

## 2 Literature Review and Related Work

Mechanical arms have been the subject of broad investigation, with various considerations centering on improving their exactness and execution. In this section, a comprehensive overview of the existing literature is provided, comparing different methods, and highlighting the unique contributions of the proposed approach [11]. Modares et al. explored the application of machine learning strategies for optimizing

mechanical arm directions [12]. Their work pointed to the progress of the effectiveness and smoothness of arm developments. Be that as it may, their approach centered on direction optimization and did not address the pivotal perspective of exactness confirmation. In our investigation, we construct upon this work by joining a virtual environment utilizing MATLAB Simulink, empowering to not as it optimized directions but perform explicit confirmation of the arm's developments.

Melchiorre et al. presented a vision-based framework for observing and following automated arm developments. By utilizing visual input, they pointed to improving the accuracy of arm situating [13]. Whereas their strategy gave meaningful experiences in real-time observation, it did not comprehensively address the challenges of exact control and confirmation. In differentiation, our proposed technique combines the plan capabilities of Fusion 360 with the reenactment control of SIMSCAPE in MATLAB Simulink. This integration authorizes the attainment of precise control, reenactment, and confirmation of the mechanical arm's developments, guaranteeing not as it were visual feedback but too thorough approval. Mohamed et al. centered on the kinematic modeling of robotic arms, pointing to get the arm's movement characteristics [14]. Their work contributed to the theoretical understanding of arm developments, giving good experiences in kinematic connections. In any case, their approach did not broadly address exactness confirmation. By combining reverse and forward kinematics, we expand prior kinematic modeling in our research (according to the literature analysis). We can achieve precise control and real-time approval of the arm's advancements thanks to this all-encompassing strategy, which ensures hypothetical precision and unwavering common sense quality.

Besides, Jhang et al. proposed a control framework that utilized input from numerous sensors to make strides in the exactness of automated arm developments [15]. Their approach illustrated upgraded exactness by joining sensor information into the control circle. In any case, expanding different sensors expanded the general framework's complexity and fetched. Our technique optimizes exactness using PID control frameworks and reverses and forward kinematics. This approach enables precise control without requiring extra sensors, guaranteeing precision enhancement while maintaining simplicity and cost-effectiveness. The proposed strategy combines past works and utilizes Fusion 360 and SIMSCAPE in MATLAB Simulink for improved precision and performance of mechanical arms. The virtual environment created enables thorough testing and reliable arm movements. In contrast to past research, this term paper suggests a novel strategy (using a virtual environment in this work) to improve the precision of robotic arms [16]. This approach emphasizes accurate direction optimization, extensive control, and thorough confirmation. Our method ensures precise and efficient arm operations in multiple industries using Fusion 360 and SIMSCAPE integration.

### 3 Proposed Method, Tools, and Techniques Used

To improve the exactness of a robotic arm in a virtual environment, we propose a comprehensive technique that combines Fusion 360 for plan and SIMSCAPE in MATLAB Simulink for reenactment and confirmation [17]. Central to our approach are the concepts of forward and reverse kinematics, which plays a significant part in empowering exact and controlled developments of the robotic arm [18].

#### 3.1 *Forward Kinematics*

Forward kinematics is the numerical representation of how joint points in a robotic arm are compared to the position and introduction of its conclusion effector within the workspace [19]. Using the geometric connections between the arm's joins and joints, forward kinematics can decide the position and introduction of the conclusion effector based on the given joint points. In our strategy, forward kinematics is vital for deciphering the required trajectories into particular joint point commands [20]. By indicating the required position and orientation of the conclusion effector, we can calculate the comparing joint points required to realize these wanted movements. This data is pivotal for controlling the robotic arm's developments precisely and absolutely.

#### 3.2 *Inverse Kinematics*

Inverse kinematics is deciding the joint points required to realize a wanted position and introduction of the conclusion effector. Unlike forward kinematics, which calculates the conclusion effector's position based on given joint points, converse kinematics agrees to work invert, deciding the joint points fundamental to reach a particular position and introduction. In our methodology, inverse kinematics plays a significant part in interpreting the required directions into joint point commands [21]. By characterizing the specified position and introduction of the conclusion effector, we utilize reverse kinematic conditions to calculate the comparing joint points required to realize these craved movements. This data shapes the premise for controlling the mechanical arm and guaranteeing its exact and exact developments [22].

The integration of forward and backward kinematics in our technique acknowledges precise control and validation of the autonomous arm growth inside the virtual environment. Using the geometrical connections between the joints and joints of the arm, be ready to decide on the final articulation points essential to perform definite directions [23]. This level of control ensures arm growth that adjusts to our planning results and improves accuracy throughout the mechanical arm. In addition, forward and backward kinematics facilitate real-time arm movements [24]. By comparing the

actual joints and concluding the position of the performance parts with the required orientations, it is possible to monitor the accuracy of the arm growth in the virtual environment [25].

This approval preparation allows us to discern inconsistencies or deviations from the specified accuracy and make fundamental changes to advance mechanical arm performance [26]. In summary, our technique's merging of forward and backward kinematics allows us to perform precise and controlled developments of the autonomous arm in a virtual environment. By leveraging the capabilities of Fusion 360 for planning and SIMSCAPE in MATLAB Simulink for reproduction and validation, we can ensure accurate execution of real-time direction and approval [27]. This approach contributes to improving the accuracy of the mechanical arm and is tuned to our goal of enhancing its performance in a virtual environment.

### 3.3 Technique

Kinematics relatively controls the movement of the robotic arm to the points of the joint with the position and introduction of the end-actuator. Both coordinate and reverse kinematics are utilized [28]. Calculate the position and introduction of the robot arm end-acting component based on joint points utilizing coordinate kinematics conditions. Coordinate kinematics condition:

$$T = A_1(q_1) \times A_2(q_2) \times A_3(q_3) \times \cdots \times A_n(q_n) \quad (1)$$

$T$  is the generator's position and introduction change framework finishing up in Cartesian space, where  $q_1, q_2, \dots$  are the factors. The points of the joints of the robot are spoken to by  $q_n$ .  $A_1, A_2, \dots, A_n$  speak to homogeneous changes of the commitment of each joint to the change of the conclusion specialist. Reverse kinematics calculates the joint points to attain the specified position and introduction of the end-acting portion, which is more complicated than forward kinematics.

$$q = f^{-1}(X) \quad (2)$$

The inverse kinematics condition is expressed as  $q$  for the joint points,  $X$  for the required position and introduction of the conclusion portion, and  $f^{-1}$  for the reverse work. Strategies like slope plunge or Jacobian relocation can unravel the issue. Forward and reverse kinematics are exceptionally imperative for robot arm development. Converse kinematics is frequently utilized when the position and introduction of the end-acting component are known. Forward kinematics decides the end-effector position and introduction based on joint points, making a difference with direction arranging and collision discovery [29]. The correct kinematics equations are exceptionally critical for the proper movement of the robot arm. Accurate calibration and reenactment are essential to specific kinematics within the situating and orientation of the conclusion channel to avoid collisions and errors. To capture

the effect of forward and inverted kinematics on robot arm movement, we ought to recognize potential deterrents, such as singularities within the kinematics condition. The singularity occurs when the robot's Jacobian lattice cannot be switched, causing it to lose control and get stuck. Disappointment to do so may harm the robot and its environment, so optimizing the robot arm's development can illuminate the issue.

The angle plummet iteratively alters the joint points to play down the distinction between the specified and actual position and the introduction of the conclusion generator, following the rule of minimizing the steepest to the least. The robot takes after the negative slant of the mistake works to attain the least utilizing the guideline of steepest plummet. It is frequently utilized for real-time control to adjust to changes within the environment or the assignment. Recreation and testing make strides in the exactness of the robot arm's development by making a virtual demonstration to assess its development [30]. Engineers can change the robot's conditions and joints before physical testing and confirming that the robot's developments are inside detail. Precision and accuracy of robot arm movements are imperative for industrial applications. The Jacobian Relocation and Slope Diminishment optimization procedures make strides in the robot's movement. Reenactment and testing confirm the exactness of the kinematic equation. If it is not too much trouble, give more settings and indicate the text you need me to shorten.

## 4 Methodology

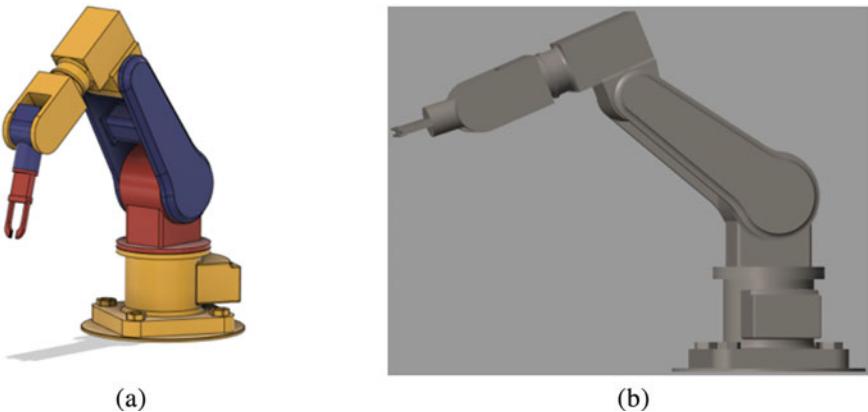
The verification of the accuracy of a robotic arm is essential to ensure that it performs its intended tasks efficiently and accurately. In this project, we will design a 6 Degree of Freedom (DOF) robotic arm using Fusion 360 and verify its accuracy using MATLAB Simulink with SIMSCAPE.

### 4.1 Design of the Robotic Arm

The first step in this project is to design the 6-DOF robotic arm in Fusion 360. The design should include all the necessary parts, including the base, links, and end-effector. The robotic arm should be able to move in all six directions, including forward, backwards, up, down, left, and right. Once the design is complete, it should be saved as a .stp file. Figure 1 shows the designed model of Proposed Design of 6-DOF Robotic Manipulator in Fusion 360.

### 4.2 Addition of Revolute Joints

After designing the robotic arm, the next step is to add three revolute joints to the model to simulate a sinusoidal path described in the desired trajectory equation. This



**Fig. 1** **a** Proposed design of 6-DOF robotic manipulator and **b** imported model after adding three revolute joints in the above robotic manipulator

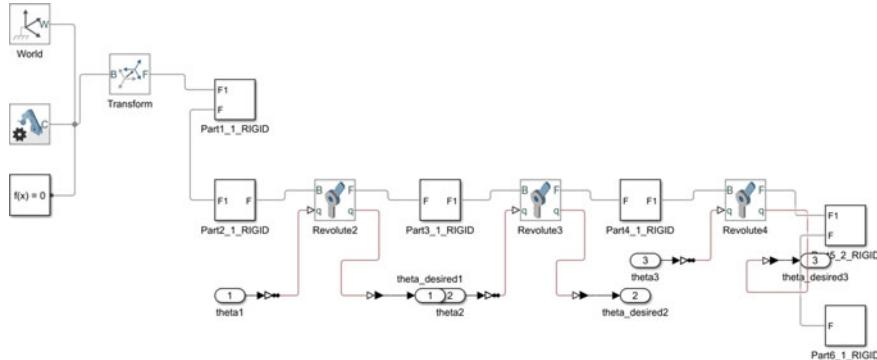
step will be done in Onshape, where the .stp file will be loaded, and the joints will be added so the robot can follow a simple direction. Once the joints are added, the model is saved again.

#### 4.3 Importing the Model to MATLAB

Once the robotic arm model is ready, it is imported into MATLAB using the SIM-SCAPE library, which converts the Onshape file to an XML file and then loads it to MATLAB. After loading the file to MATLAB, it is opened in SIMULINK, where the robot environment is defined.

#### 4.4 Defining the Robot Environment

The robot environment is defined by adding three input and output blocks connected to a Simulink to a physical converter block operated as a second-order derivative. This block is then joined with the revolute joint block, and the output of this block is connected to the physical Simulink converter block, which is further connected with the output block. This process is done for all three revolute joints. Once this is done, the entire system is grouped into a subsystem. Figure 2 shows a Block diagram of robotic manipulator constructed in MATLAB SIMULINK environment representing the 3 Revolute joints and the input taken in the form of angles to the motors controlling the three joints.



**Fig. 2** Block diagram of robotic manipulator constructed in MATLAB SIMULINK

#### 4.5 Defining the Desired Trajectory

To verify the precision of the robotic arms, a requested path is defined using the trajectory equation. A unit signal is taken from a clock, which gives input to the desired trajectory block, from which three equations are generated  $x_d$ ,  $y_d$ , and  $z_d$ . One input is given to the scope, which generates the graphs of the desired trajectory, and the second is taken as input to the inverse kinematic block. The below equation is considered the desired trajectory equation as follows:

$$\text{function}[x_d, y_d, z_d] = \text{Desired Trajectory}(u) \quad (3)$$

$$x_d = 1 + 0.5 \sin\left(\frac{\pi}{4}u\right); \quad y_d = 1 + 0.5 \cos\left(\frac{2\pi}{3}u\right); \quad z_d = 1 - 0.5 \sin\left(\frac{\pi}{3}u + \frac{\pi}{2}\right) \quad (4)$$

#### 4.6 Working of Inverse Kinematic Block

Using inverse kinematic equations, the inverse kinematic block generates  $\theta_{1d}$ ,  $\theta_{2d}$ , and  $\theta_{3d}$  which are then given as input to the three different sum blocks and the three different PID blocks. The output of these blocks is given to the grouped subsystem, where the three input blocks take the input of  $\theta_{1d}$ ,  $\theta_{2d}$ , and  $\theta_{3d}$  and produce  $\theta_{1\text{desired}}$ ,  $\theta_{2\text{desired}}$ , and  $\theta_{3\text{desired}}$ . The below equations are considered the Inverse Kinematic equation for getting  $\theta_{1d}$ ,  $\theta_{2d}$ , and  $\theta_{3d}$ .  $l_1$ ,  $l_2$ , and  $l_3$  represent the lengths of the links or segments in a robotic arm or manipulator.

$$\text{function}[\theta_{1d}, \theta_{2d}, \theta_{3d}] = \text{inverse}_{\text{kinematics}}(x_d, y_d, z_d) \quad (5)$$

$$l_1 = 2; \quad l_2 = 2; \quad l_3 = -2 \quad (6)$$

$$r = \sqrt{xd^2 + yd^2}; s = zd - l_1; D = \frac{r^2 + s^2 - l_2^2 - l_3^2}{2 \cdot l_2 \cdot l_3} \quad (7)$$

if  $abs(D) > 1$  error (The desired position is out of reach)

$$\theta_{3d} = \tan^{-1} 2 \left( \frac{\sqrt{1 - D^2}}{D} \right) \quad (8)$$

$$\theta_{2d} = \tan^{-1} 2 \left( \frac{s}{r} \right) - \tan^{-1} 2 \left( \frac{(l_3 \sin(\theta_{3d}))}{l_2 + l_3 \cos(\theta_{3d})} \right) \quad (9)$$

$$\theta_{1d} = \tan^{-1} \left( \frac{y_d}{x_d} \right) \quad (10)$$

#### 4.7 Working of Forward Kinematic Block

The output of this block is used for feedback providence and as an input for the Forward kinematic block, which, with the help of Forward Kinematic equations, generates actual theta values, which are then given to the scope block to compare the change. This step is crucial in verifying the correctness of robotic arms. The below equation stands as the Forward Kinematic equation for getting  $xa$ ,  $ya$ ,  $za$ :

$$\text{function}[xa, ya, za] = \text{Forward}_{\text{Kinematics}}(\theta_1a, \theta_2a, \theta_3a) \quad (11)$$

$$x_a = \cos(\theta_1a) \cdot (l_2 \cdot \cos(\theta_2a) + l_3 \cdot \cos(\theta_2a + \theta_3a)) \quad (12)$$

$$y_a = \sin(\theta_1a) \cdot (l_2 \cdot \cos(\theta_2a) + l_3 \cdot \cos(\theta_2a + \theta_3a)) \quad (13)$$

$$z_a = l_1 + l_2 \cdot \sin(\theta_2a) + l_3 \cdot \sin(\theta_2a + \theta_3a) \quad (14)$$

#### 4.8 Data Collection and Analysis

Once the robotic arm has been tried, information is collected from the scope piece, and the yield values are analyzed. The examination should incorporate comparing the theta points' required and actual values. The mechanical arm is considered precise if the distinction between the two values is satisfactory. In conclusion, the exactness of a 6 Degree of Opportunity mechanical arm was confirmed utilizing Fusion 360, Onshape, MATLAB Simulink, and SIMSCAPE. The Proposed Block diagram model of the operating manipulator takes desired trajectories as input and provides motion based on the input equations is shown in Fig. 3.

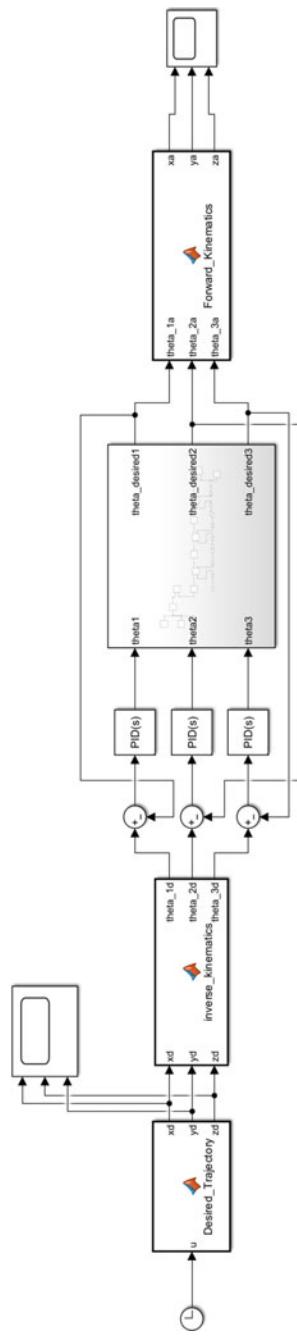


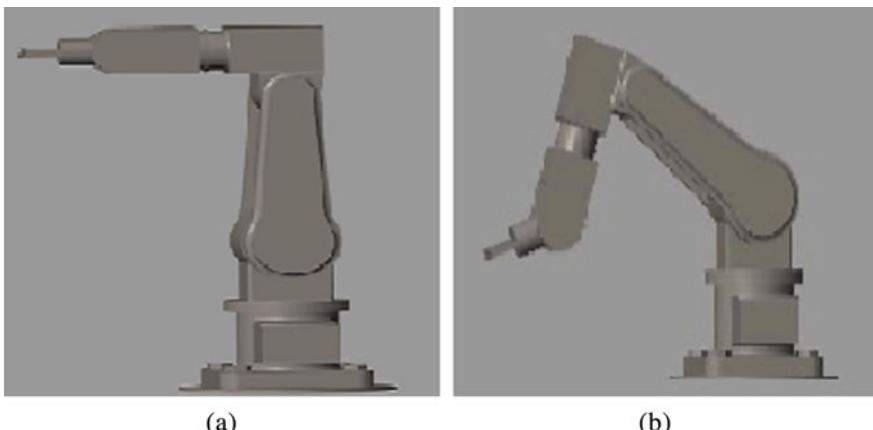
Fig. 3 Proposed Block diagram model of the operating manipulator

The plan of the mechanical arm was made in Fusion 360, and revolute joints were included in Onshape to mimic a sinusoidal way. The show was then imported to MATLAB utilizing SIMSCAPE, where the robot environment was characterized by adding input and yield pieces. The required direction was characterized utilizing the direction condition, and the converse kinematic piece produced theta values. At that point, these values were utilized as inputs for the forward kinematic square, which created actual theta values. At last, information was collected and analyzed to compare the specified and actual values of the theta points to confirm the exactness of the robotic arm. The method illustrated in this extension can be connected to confirm the precision of other automated arm plans.

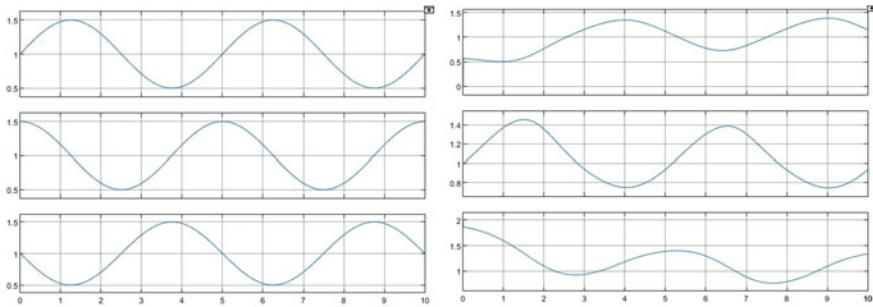
## 5 Result

The extend reenacts a 3-link automated controller get-together on MATLAB Simulink utilizing SIMSCAPE. The exactness of the robotic arm is affirmed by comparing the specified direction with the gotten direction after utilizing all the reverse and forward kinematics thoughts. Figures 4a and 5 shows Initial movement provided by the input equation and displays the desired trajectory's sinusoidal graph respectively.

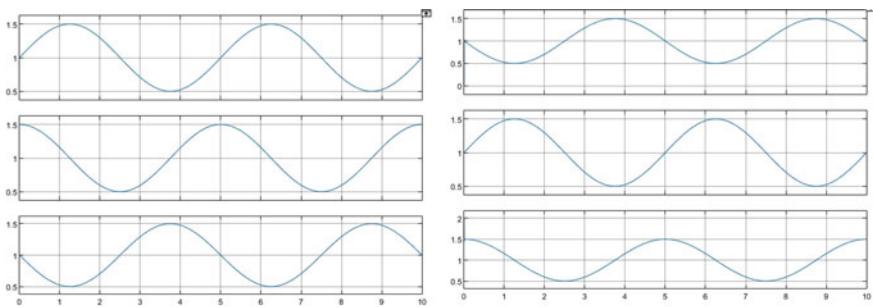
It has tended to how converse and forward kinematics influence the robotic arm's movement and how PID controllers oversee the development of the engines. The exactness of the movement depends on the tuning of the PID controller, and any changes made to the parameters, such as the accuracy of the mechanical arm's sinusoidal direction way can be affected by the length of the joins, the mass of the end-effector, contact at the joints, and PID controller tuning. In conclusion, this ven-



**Fig. 4** **a** Initial movement provided by the input equation. **b** Final position and optimized path obtained from the input equation after tuning the PID controller



**Fig. 5** Graphical representation of Initial movement obtained by the input equation



**Fig. 6** Graphical representation of Final position and optimized path obtained from the input equation after tuning the PID controller

ture gives good knowledge into robotic arms' plan and control utilizing converse and forward kinematics and PID controller. It can be expanded to more complex mechanical arm plans and control frameworks. Below are the charts of the required vs real direction sometime recently PID Tuning. Figures 4b and 6 shows final position and optimized path with graphical representation obtained from the input equation after tuning the PID controller which moves the manipulator more precisely and effectively.

## 6 Conclusion

In conclusion, the kinematics of a mechanical arm plays a pivotal part in its development and exactness. Forward kinematics is utilized to calculate the position and introduction of the end-effector for a given set of joint points, whereas reverse kinematics is utilized to calculate the points of the robot's joints that will result in a craved end-effector position and introduction. The precision of these conditions is fundamental for adequately developing the robotic arm, and blunders can lead to

collisions or other issues. Optimization procedures like slope plunge and Jacobian transpose can move forward the precision and exactness of the robotic arm's development to address challenges such as singularities and joint limits. Recreation and testing can be utilized to confirm the exactness of the kinematic conditions recently actualizing them on the physical robot. To this extent, a 6-DOF mechanical arm was outlined utilizing Fusion 360, and its precision was confirmed utilizing MATLAB Simulink with SIMSCAPE. By carefully planning and confirming the mechanical arm's kinematics, we can guarantee that it performs its planning errands productively and precisely in real-world mechanical applications.

## References

1. Vachtsevanos G, Davey K, Lee KM (1987) Development of a novel intelligent robotic manipulator. *IEEE Control Syst Mag* 7(3):9–15
2. Miyamoto Hiroyuki, Kawato Mitsuo, Setoyama Tohru, Suzuki Ryoji (1988) Feedback-error-learning neural network for trajectory control of a robotic manipulator. *Neural Netw* 1(3):251–265
3. Billard A, Kragic, D (2019) Trends and challenges in robot manipulation. *Science* 364(6446):eaat8414
4. Piltan F, Sulaiman NB (2012) Review of sliding mode control of robotic manipulator. *World Appl Sci J* 18(12):1855–1869
5. Ouyang PR, Zhang WJ, Gupta MN (2006) An adaptive switching learning control method for trajectory tracking of robot manipulators. *Mechatronics* 16(1):51–61
6. Baek J, Cho S, Han S (2017) Practical time-delay control with adaptive gains for trajectory tracking of robot manipulators. *IEEE Trans Ind Electron* 65(7):5682–5692
7. Zhu M, Ye L, Ma X (2020) Estimation-based quadratic iterative learning control for trajectory tracking of robotic manipulator with uncertain parameters. *IEEE Access* 8:43122–43133
8. Yang X, Feng Z, Liu C, Ren X (2014) A geometric method for kinematics of delta robot and its path tracking control. In: 2014 14th International conference on control, automation and systems (ICCAS 2014). IEEE, pp 509–514
9. Kumar RR, Chand P (2015) Inverse kinematics solution for trajectory tracking using artificial neural networks for SCORBOT ER-4u. In: 2015 6th International conference on automation, robotics and applications (ICARA). IEEE, pp 364–369
10. Dastgerdi HR, Keshmiri M (2010) Design of length measuring system for stewart platform using new forward kinematics solution. In: 2010 11th International conference on control automation robotics & vision. IEEE, pp 2339–2344
11. Bilal H, Yin B, Aslam MS, Anjum Z, Rohra A, Wang Y (2023) A practical study of active disturbance rejection control for rotary flexible joint robot manipulator. *Soft Comput*, pp 1–15
12. Modares H, Ranatunga I, Lewis FL, Popa DO (2015) Optimized assistive human–robot interaction using reinforcement learning. *IEEE Trans Cybern* 46(3):655–667
13. Melchiorre M, Scimmi LS, Mauro S, Pastorelli SP (2021) Vision-based control architecture for human–robot hand-over applications. *Asian J Control* 23(1):105–117
14. Mohamed NA, Azar AT, Abbas NE, Ezzeldin MA, Ammar HH (2020) Experimental kinematic modeling of 6-dof serial manipulator using hybrid deep learning. In: Proceedings of the international conference on artificial intelligence and computer vision (AICV2020). Springer, pp 283–295
15. Jhang LH, Santiago C, Chiu CS (2017) Multi-sensor based glove control of an industrial mobile robot arm. In: 2017 International automatic control conference (CACS). IEEE, pp 1–6

16. Srisuk P, Sento A, Kitjaidure Y (2017) Inverse kinematics solution using neural networks from forward kinematics equations. In: 2017 9th International conference on knowledge and smart technology (KST). IEEE, pp 61–65
17. Srisuk P, Sento A, Kitjaidure Y (2017) Forward kinematic-like neural network for solving the 3d reaching inverse kinematics problems. In: 2017 14th International conference on electrical engineering/electronics, computer, telecommunications and information technology (ECTI-CON). IEEE, pp 214–217
18. Iqbal J, Islam RU, Khan H et al (2012) Modeling and analysis of a 6 dof robotic arm manipulator. *Can J Electr Electron Eng* 3(6):300–306
19. Ahmed A, Yu M, Chen F (2022) Inverse kinematic solution of 6-dof robot-arm based on dual quaternions and axis invariant methods. *Arab J Sci Eng* 1–16
20. Karlik B, Aydin S (2000) An improved approach to the solution of inverse kinematics problems for robot manipulators. *Eng Appl Artif Intell* 13(2):159–164
21. Khan H, Kim HH, Abbasi SJ, Lee MC (2020) Real-time inverse kinematics using dual particle swarm optimization dspo of 6-dof robot for nuclear plant dismantling. *IFAC-PapersOnLine* 53(2):9885–9890
22. Alkayyali M, Tutunji TA (2019) Pso-based algorithm for inverse kinematics solution of robotic arm manipulators. In: 2019 20th international conference on research and education in mechatronics (REM). IEEE, pp 1–6
23. Megalingam RK, Katta N, Geesala R, Yadav PK, Rangaiah RC (2018) Keyboard-based control and simulation of 6-dof robotic arm using ros. In: 2018 4th International conference on computing communication and automation (ICCCA). IEEE, pp 1–5
24. Schwarz A, Höller MK, Pereira J, Ofner P, Müller-Putz GR (2020) Decoding hand movements from human EEG to control a robotic arm in a simulation environment. *J Neural Eng* 17(3):036010
25. Arleo G, Caccavale F, Muscio G, Pierri F (2013) Control of quadrotor aerial vehicles equipped with a robotic arm. In: 21St Mediterranean conference on control and automation. IEEE, pp 1174–1180
26. Kostic D, De Jager B, Steinbuch M, Hensen R (2004) Modeling and identification for high-performance robot control: an RRR-robotic arm case study. *IEEE Trans Control Syst Technol* 12(6):904–919
27. Aliff M, Dohta S, Akagi T (2015) Trajectory control of robot arm using flexible pneumatic cylinders and embedded controller. In: 2015 IEEE International conference on advanced intelligent mechatronics (AIM). IEEE, pp 1120–1125
28. Huang Z, Li F, Xu L (2020) Modeling and simulation of 6 dof robotic arm based on gazebo. In: 2020 6th International conference on control, automation and robotics (ICCAR). IEEE, pp 319–323
29. Bi M (2020) Control of robot arm motion using trapezoid fuzzy two-degree-of-freedom PID algorithm. *Symmetry* 12(4):665
30. Ohta P, Valle L, King J, Low K, Yi J, Atkeson CG, Park YL (2018) Design of a lightweight soft robotic arm using pneumatic artificial muscles and inflatable sleeves. *Soft Robot* 5(2):204–215

# Human Activity Recognition a Comparison Between Residual Neural Network and Recurrent Neural Network



K. P. Anu and J. V. Bibal Benifa

**Abstract** Recognizing human activity is important for interpersonal interactions and human-to-human communication. It is challenging to extract since it contains details about a person's identity, personality, and psychological condition. One of the key research topics in the fields of computer vision and machine learning is the human capacity for activity recognition. This study has led to the need for a multimodal activity identification system in several applications, such as video surveillance systems, human-computer interaction, and robots for characterizing human behavior. In this study, Residual Neural Network (ResNet50) and Recurrent Neural Network (RNN) architectures for human activity recognition (HAR) are compared. Our assessment is based on a number of performance metrics, such as accuracy F1-score, recall, and computational effectiveness. Here we have the model accuracy of ResNet50 at 53% and RNN at 23% with epoch 10. Our objective is to evaluate how well the models can categorize various human activities. The outcomes demonstrate that ResNet50 outperformed RNN.

**Keywords** Computer vision · HAR · CNN · ResNet50 · RNN

## 1 Introduction

HAR describes the method of automatically recognizing and categorizing human actions using information gathered from a variety of sensors or input sources. The purpose of HAR is to understand and interpret human actions in real-time or from previous records using machine learning and pattern recognition algorithms. There are several HAR uses in numerous fields. Healthcare and wellbeing: HAR can be used in the healthcare industry to track rehabilitation progress, identify falls, monitor patients' physical activity levels, and offer individualized feedback and exercise sug-

---

K. P. Anu (✉) · J. V. Bibal Benifa  
Indian Institute of Information Technology, Kottayam, India  
e-mail: [anu.phd2103@iiitkottayam.ac.in](mailto:anu.phd2103@iiitkottayam.ac.in)

J. V. Bibal Benifa  
e-mail: [benifa@iiitkottayam.ac.in](mailto:benifa@iiitkottayam.ac.in)

gestions. Helping elderly people with daily tasks, as well as enhancing their safety, wellbeing, and autonomy, are currently seen as important research topics of considerable interest [1]. With applications in robot vision, multimedia content search, video surveillance, and motion tracking systems, human action recognition (HAR) has recently attracted the attention of computer vision researchers also in additionally, recent advancements in artificial intelligence have prompted computer vision researchers to look into issues with recognizing actions [2]. Sports and fitness: HAR may be used to assess performance, examine movement patterns, and offer training improvement insights for athletes and fitness aficionados. Additionally, it can be used in sports coaching to give in-the-moment feedback and improve technique. Smart Homes and Assistive Technologies: HAR can make smart home systems responsive to users' actions by automatically altering lighting, temperature, and other settings in accordance with identified activities. It may also be used to create assistive technologies that enable persons with impairments to use equipment and communicate with their surroundings using gesture-based commands. Security and surveillance: HAR can be used in video surveillance systems to identify unauthorized entry, theft, or violent behavior. It can aid in seeing possible dangers and setting off the right reactions. Early care: HAR may help with care for the elderly by keeping an eye on their activities and wellbeing. It can spot changes from the norm or spot signals of distress, allowing for quick action or notifications to carers or medical staff. Context-Aware Systems: By supplying contextual data about a person's behaviors, HAR enables context-aware systems to provide more specialized and pertinent services. A smartphone, for instance, may automatically modify its behavior based on the detected activity, such as turning off alerts when in a conference or recommending local eateries while it is lunchtime. These are just a few instances of the many sectors in which human activity recognition may be used. As technology develops, new opportunities and uses will become possible in the future. Several studies have proposed approaches for detecting human activity using machine learning methods and various kinds of sensors [3]. The idea of HAR is not new, and numerous studies have been done in this area. However, older machine learning techniques, which required handcrafted features engineering and had poorer accuracy, are the main focus of the present literature. Additionally, without doing a thorough examination, some writers proposed deep learning-based HAR systems by transferring these techniques from other domains directly to the HAR domain. The primary focus of the current deep learning-based techniques is CNN [4]. Deep learning models are used for diabetic retinopathy detection, skin cancer classification and early detection, COVID-19 detection, face detection, etc. [5–8].

A deep learning model known as a convolutional neural network (CNN) is frequently used for a variety of computer vision applications, such as image classification, object identification, and image segmentation [9]. Deep learning is a general term for neural networks that use many layers of hierarchically arranged non-linear information processing for feature extraction and classification, with each layer processing the outputs of the previous layer [10]. It has been created to effectively handle and analyze visual data by using the spatial correlations present in images. Here we compare the performance of ResNet50 and RNN models. The neural net-

work, which functions as a black box and accurately models the problem, may be given data directly. This indicates that the neural networks can accurately identify the type of movement correctly [11].

The models are evaluated on the testing set, and the classification report provides insights into the model's performance for each activity class. Precision, recall, and F1-score are frequently used as classification evaluation metrics to evaluate a machine learning model's performance [12]. Precision is the measure of the model's accuracy in predicting positive samples correctly. It is calculated as the ratio of true positive predictions to the total number of positive predictions. Precision represents how well the model avoids false positives. A higher precision indicates fewer false positive predictions.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (1)$$

False Negatives (FN) is the number of missed positive predictions. The F1-score is a metric that combines both precision and recall into a single value. The harmonic mean of precision and recall provides a balanced measure of the model's performance. The F1-score considers false positives and negatives and is especially useful when the dataset is imbalanced. A high F1-score indicates a model that achieves a good balance between precision and recall.

$$\text{F1-score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (2)$$

Support refers to the number of samples in each class in the evaluation set. It represents the distribution of the actual classes and provides context for interpreting precision, recall, and F1-score. Support is crucial for understanding the performance of the model across different classes and identifying potential biases or imbalances in the data. A deep learning model typically comprises a few thousand to millions of parameters, which makes direct development of scientific insight and model interpretation challenging [13].

For experimental purposes, we collected the dataset from the Kaggle open dataset repository (<https://www.kaggle.com/datasets/shashankrapolu/human-action-recognition-dataset>). In this dataset, the training folder contains 10710 images belonging to 15 classes, and the testing folder contains 1890 images belonging to 15 classes. The different classes, which include 'calling', 'clapping', 'cycling', 'dancing', 'drinking', 'eating', 'fighting', 'hugging', 'laughing', 'listening\_to\_music', 'running', 'sitting', 'using\_laptop' 'sleeping', and 'texting' [14].

The rest of this paper is organized as follows: Sect. 2 gives the algorithm and implementation results of ResNet50. Section 3 gives the algorithm and implementation results of RNN. Finally, the paper concludes by giving its conclusion and expecting future works.

## 2 HAR Using ResNet50

Numerous techniques have been proposed as a result of the extensive study that has been done in recent years to examine various sensing technologies for simulating and identifying human activity [15]. Deeper convolutional neural networks are typically more challenging to train. As the layer depth grows, the model's accuracy typically decreases and ResNet50, on the other hand, are simpler to optimize and can improve accuracy as depth grows [16]. ResNet50 is Residual Network and it is a deep neural network architecture known for its ability to train very deep networks effectively by addressing the vanishing gradient problem. It introduces skip connections or residual connections that allow gradients to flow directly through the network, enabling the training of deeper models. The ResNet50 architecture consists of several residual blocks, where each block contains multiple convolutional layers. Each residual block has a 'skip connection' that bypasses some convolutional layers and adds the input of the block to the output [17]. This skip connection helps to propagate gradients effectively and makes it easier to learn identity mapping. We implemented the ResNet50 model using Keras Sequential API. It starts with a Convolutional layer with 64 filters and a kernel size of (7, 7), followed by a MaxPooling layer. This is followed by four stages, each containing multiple residual blocks. The number of filters in each stage gradually increases, while the spatial dimensions decrease due to the MaxPooling layers. Within each stage, the residual blocks consist of multiple convolutional layers with  $3 \times 3$  filters. The number of filters doubles after each downsampling operation (MaxPooling layer) to maintain the same number of channels throughout the blocks. The ResNet50 network may take an input image with a height, width, and channel width that are multiples of 32 [18]. The skip connection connects the input to the output of each block, allowing the gradients to propagate effectively. In the last stage, a GlobalAveragePooling layer is used to reduce the spatial dimensions to a vector. This is followed by a Dense layer with 256 units and a ReLU activation function. Finally, a Dense layer with the number of classes as units and a softmax activation function is added to obtain the final class probabilities. Here we compiled the model with the Adam optimizer and sparse categorical cross-entropy loss function, which is suitable for multi-class classification problems. It is then trained using the fit() function, specifying the training data ( $X_{\text{train}}$  and  $y_{\text{train}}$ ), batch size, number of epochs, and validation data ( $X_{\text{test}}$  and  $y_{\text{test}}$ ). During training, the model learns to optimize its parameters by minimizing the loss function and adjusting the weights of the convolutional layers.

Images go through preprocessing procedures, such as scaling, normalization, rotation, before being fed into the ResNet50 model. Rectified Linear Unit (ReLU) activation function is used by the ResNet50 model to introduce non-linearity and learn complex patterns in the data. Applying learnable filters to the input data, creating feature maps, and then using the activation function are the main operations of the convolution layer. It is possible to regulate the output spatial dimensions using stride and padding. The evaluation metrics such as accuracy, loss, precision, recall, and F1-score are computed during training to monitor the model's performance as mentioned in Fig. 1. The algorithm for ResNet50 can be summarized as follows:

60/60 [=====] - 181s 3s/step				
	precision	recall	f1-score	support
0	0.44	0.56	0.49	126
1	0.48	0.52	0.50	126
2	1.00	0.59	0.74	126
3	0.54	0.71	0.61	126
4	0.49	0.42	0.45	126
5	0.86	0.76	0.81	126
6	0.87	0.37	0.51	126
7	0.67	0.34	0.45	126
8	0.79	0.52	0.63	126
9	0.40	0.41	0.41	126
10	0.57	0.66	0.61	126
11	0.54	0.40	0.46	126
12	0.89	0.45	0.60	126
13	0.25	0.67	0.37	126
14	0.46	0.60	0.52	126
accuracy				0.53
macro avg				0.54
weighted avg				0.54
				1890

**Fig. 1** Classification report of ResNet50

1. Input: RGB image of size  $(224 \times 224 \times 3)$ .
2. Convolutional layers:
  - Convolutional layer with  $7 \times 7$  kernel, stride 2, and 64 filters.
  - Batch normalization and ReLU activation.
  - Max pooling with  $3 \times 3$  pool size and stride 2.
3. Residual blocks:
  - Four stages, each containing multiple residual blocks.
  - Each stage has a different number of residual blocks and filter sizes.
  - Each residual block consists of multiple convolutional layers with  $1 \times 1$ ,  $3 \times 3$ , and  $1 \times 1$  kernels.
  - Batch normalization and ReLU activation after each convolutional layer.
  - Skip connections that add the original input to the output of the residual block.
4. Global average pooling:
  - Average pooling across the spatial dimensions of the feature map.
5. Fully connected layers:
  - Flatten the output from the previous step.
  - Dense layer with 1000 units and softmax activation for ImageNet classification.

## 6. Training:

- Initialize the model with pre-trained weights on ImageNet or random weights.
- Fine-tune the model on the target task (e.g., human action recognition) using labeled training data.
- Use an optimizer (e.g., stochastic gradient descent) to minimize the loss function.
- Update the model's weights based on the gradients computed during backpropagation.
- Repeat the training process for a certain number of epochs or until convergence.

## 7. Evaluation:

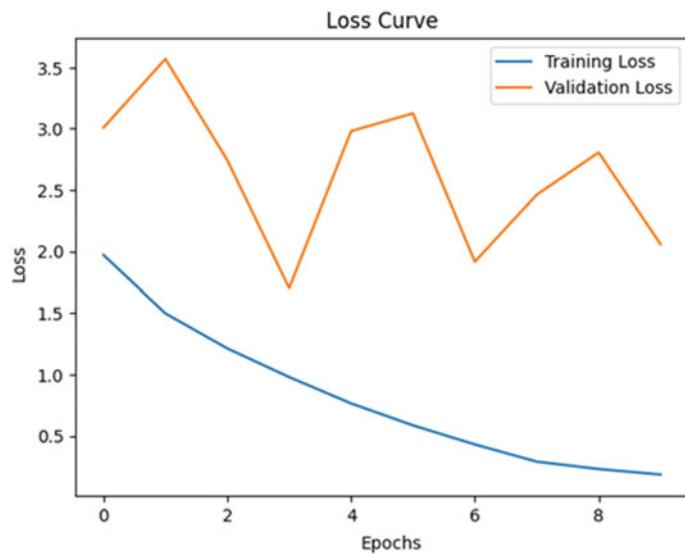
- Evaluate the trained model on a separate testing dataset.
- Compute the predicted labels for the testing samples.
- Compare the predicted labels with the ground truth labels.
- Calculate metrics such as accuracy, precision, recall, and F1-score to assess the model's performance.

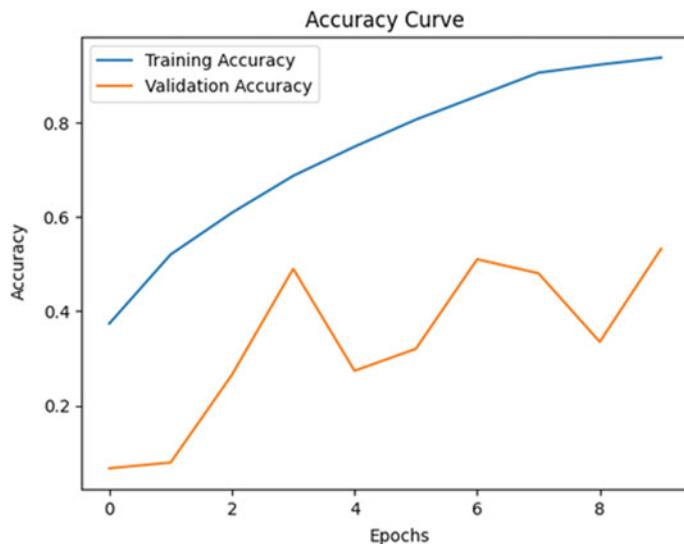
So based on the class probabilities we can predict the output label based on the highest probability value.

The ResNet50 model for HAR can perform differently under different conditions and scenarios like dataset size and diversity, image quality, the model complexity and depth, class imbalance, etc. If the dataset is small then our model struggle to accurately predict the activities, also our images are less of quality like less resolution which will also affect the accuracy of our model. Using a complicated model like ResNet50 can result in overfitting if the dataset is small. The effectiveness of the model can also be impacted by a class imbalance in the HAR dataset, particularly if particular behaviors are underrepresented.

ResNet50 we can use in different applications such as fall detection systems, which seek to identify and categorize falls from video or depth sensor data. In order to separate falls from regular activity, the model can acquire discriminative properties relating to human body movements, postures, and dynamics. Intricate patterns can help with effective fall detection, which is critical for elderly care and safety applications. It also has been used in sports activity analysis with the aim of identifying and analyzing the actions and movements of athletes during various sporting events. The model can be trained on picture or video data to recognize actions like dribbling, shooting, passing, or other sports methods. As a result of ResNet50's deep architecture, precise sports activity analysis is made possible by the capture of fine-grained motion patterns and poses.

		Confusion Matrix															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
True Labels	0	71	3	0	1	3	0	0	0	3	12	2	0	0	16	15	
	1	4	66	0	7	5	3	1	1	4	6	1	6	0	18	4	
	2	0	0	74	10	1	0	0	1	0	10	18	8	1	2	1	
	3	4	5	0	89	2	0	1	0	0	4	11	1	0	8	1	
	4	12	10	0	3	53	2	0	1	6	3	1	4	0	28	3	
	5	2	8	0	0	3	96	0	0	1	1	0	1	0	12	2	
	6	5	9	0	25	3	1	46	1	0	2	12	8	2	12	0	
	7	12	9	0	1	8	2	1	43	1	6	2	1	2	28	10	
	8	4	11	0	1	6	1	0	4	66	4	1	1	0	20	7	
	9	15	1	0	2	8	0	0	0	1	52	2	1	1	34	9	
	10	4	6	0	18	0	0	3	0	0	1	83	4	0	5	2	
	11	10	2	0	4	5	5	0	0	1	8	6	50	0	19	16	
	12	3	1	0	4	2	1	1	10	1	9	3	1	57	20	13	
	13	7	4	0	1	9	0	0	2	0	8	2	2	0	85	6	
	14	9	3	0	0	1	1	0	1	0	3	1	4	1	27	75	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14		

**Fig. 2** Confusion matrix of ResNet50**Fig. 3** Loss curve of ResNet50



**Fig. 4** Accuracy curve of ResNet50



**Fig. 5** Sample input image of ResNet50 model [14]

### 3 HAR Using RNN

A neural network type known as an RNN introduces the idea of recurrent connections. One of the other RNN strategies that can address the problem that has gained popularity is the feed forward network, which is a deep learning model that is also utilized for sequential tasks and these networks are particularly effective at gathering

```

Image: 3
True Label: drinking
Predicted Label: drinking
Class Probabilities:
calling: 6.846678297733888e-05
clapping: 4.136603365623159e-06
cycling: 0.0027684250380843878
dancing: 6.984014180488884e-05
drinking: 0.4129810333251953
eating: 1.371875271161116e-07
fighting: 8.40004940982908e-06
hugging: 0.2085353136062622
laughing: 0.003053388325497508
listening_to_music: 0.37213799357414246
running: 0.00010730664507718757
sitting: 2.2438189262175e-05
sleeping: 0.00013484733062796295
texting: 0.00010773956455523148
using_laptop: 4.699992643963924e-07

```

**Fig. 6** Predicted label and probabilities of different classes of ResNet50 model

both sequential and temporal data [19]. RNN has a hidden state that is information is passed from one step to the next. The hidden state functions as a memory that stores data from earlier inputs and affects predictions made in the future.

RNN architecture has been modified to include long short-term memory networks (LSTM). It introduces new memory cells with a more intricate structure called LSTM cells. An input gate, a forget gate, and an output gate are the three major parts of an LSTM cell. Information entering the memory cell is controlled by the input gate, information being deleted from the memory cell is controlled by the forget gate, and information being output from the cell is controlled by the output gate. With the help of this gated structure, LSTM cells may update and transmit information selectively over time, which improves their ability to detect long-term relationships. The algorithm for RNN can be summarized as follows:

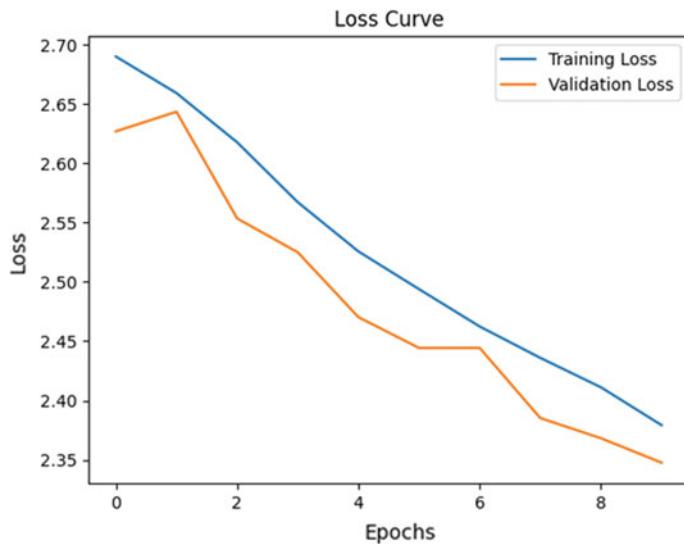
1. Load the training images and labels from the training set folder.
2. Load the testing images and labels from the testing set folder.
3. Preprocess the images:
  - Resize the images to the desired input shape (e.g., (64, 64, 3)).
  - Normalize the pixel values (e.g., scale them between 0 and 1).

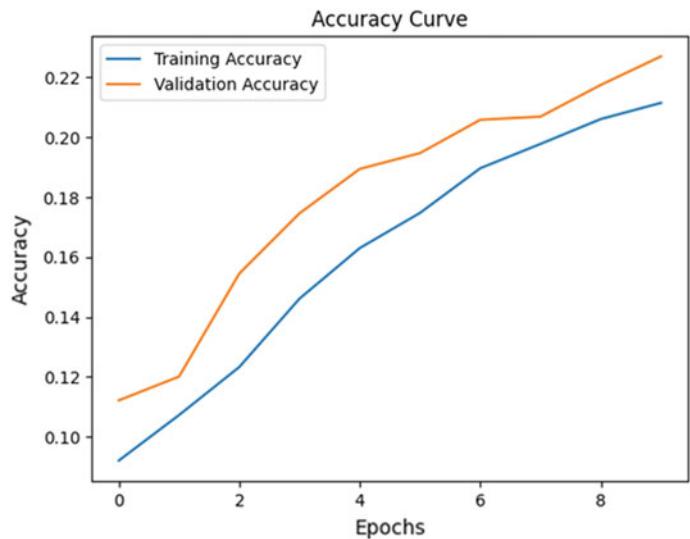
60/60 [=====] - 2s 31ms/step				
	precision	recall	f1-score	support
0	0.10	0.06	0.07	126
1	0.16	0.09	0.11	126
2	0.36	0.37	0.36	126
3	0.33	0.31	0.32	126
4	0.19	0.08	0.11	126
5	0.21	0.65	0.31	126
6	0.29	0.42	0.35	126
7	0.11	0.06	0.07	126
8	0.16	0.18	0.17	126
9	0.24	0.13	0.16	126
10	0.19	0.13	0.15	126
11	0.21	0.16	0.18	126
12	0.28	0.45	0.35	126
13	0.09	0.02	0.03	126
14	0.20	0.31	0.24	126
accuracy				0.23
macro avg				0.20
weighted avg				0.20
				1890

**Fig. 7** Classification report of LSTM

4. Perform label encoding on the labels:
  - Convert the categorical labels into numerical representations.
5. Reshape the input data to match the RNN input shape:
  - Determine the number of samples, number of frames, height, and width of the images.
  - Reshape the images into a 3D tensor with shape (num\_samples, num\_frames, height \* width).
6. Create the RNN model:
  - Use the Sequential API from Keras.
  - Add an LSTM layer with a specified number of units, input shape, and return sequences parameter.
  - Optionally, add additional LSTM layers for deeper architecture.
  - Add Dense layers with appropriate activation functions.
  - Add Dropout layers to prevent overfitting (optional).
7. Compile the model:
  - Specify the optimizer, loss function, and evaluation metrics.

Confusion Matrix															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
True Labels	7	5	2	8	5	14	5	3	15	18	1	6	13	2	22
0	5	11	1	6	4	30	12	6	9	3	1	7	12	0	19
1	1	0	47	8	0	11	22	2	0	0	23	8	2	0	2
2	4	4	4	39	0	30	14	2	1	7	5	2	10	1	3
3	5	3	1	3	10	20	2	6	16	4	5	8	17	2	24
4	0	5	2	5	1	82	4	2	9	0	4	4	4	0	4
5	0	7	12	10	0	23	53	6	0	1	8	2	2	1	1
6	9	3	7	4	5	28	5	7	21	2	6	6	11	0	12
7	4	6	2	6	2	31	8	7	23	5	1	3	16	4	8
8	8	8	1	6	6	17	6	2	17	16	4	7	11	2	15
9	1	1	35	7	1	22	12	4	7	0	16	5	4	2	9
10	4	8	11	6	3	17	14	4	4	3	4	20	10	3	15
11	2	3	2	6	1	21	9	2	4	2	3	4	57	3	7
12	6	2	4	3	6	30	10	5	10	4	5	4	19	2	16
13	11	4	1	1	9	19	4	4	4	3	0	11	16	0	39
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

**Fig. 8** Confusion Matrix of LSTM**Fig. 9** Loss curve of LSTM



**Fig. 10** Accuracy curve of LSTM



**Fig. 11** Sample input image of LSTM model [14]

#### 8. Train the model:

- Fit the model to the training data.
- Specify the batch size, number of epochs, and validation data.

```
Image: 4
True Label: clapping
Predicted Label: clapping
Class Probabilities:
calling: 0.05361596867442131
clapping: 0.21796023845672607
cycling: 0.02469111792743206
dancing: 0.04992680624127388
drinking: 0.11172199249267578
eating: 0.07393398880958557
fighting: 0.04630054160952568
hugging: 0.06163563206791878
laughing: 0.058821868151426315
listening_to_music: 0.044047340750694275
running: 0.017993677407503128
sitting: 0.09004953503608704
sleeping: 0.02615794725716114
texting: 0.050005074590444565
using_laptop: 0.0731382742524147
```

**Fig. 12** Predicted label and probabilities of different classes of LSTM model

#### 9. Evaluate the model:

- Use the test data to evaluate the model's performance.
- Calculate metrics such as accuracy, precision, recall, and F1-score.
- Optionally, save the trained model for future use.

#### 10. Save the model

For ResNet50 and LSTM models we used a batch size of 32 and the number of epochs is 10. The optimizer used here is 'adam'(Adaptive Moment Estimation) is an adaptive optimization technique that may be used to solve a variety of optimization problems and is used to train deep learning models. Adam effectively manages sparse gradients and speeds up convergence by changing the learning rates of each parameter separately. Additionally, it uses bias correction to increase the precision of moving averages and momentum to smooth out the optimization process. Adam is extensively used and, overall, has better performance. We executed these codes in Google Colab Pro and its having 25GB RAM and K80, P100, T4 GPU. After fitting the model we plotted the classification report, confusion matrix, loss curve, accuracy curve, random input image, and different class probabilities. All the plots of ResNet50 are shown in Figs. 1, 2, 3, 4, 5, and 6 respectively. In the case of LSTM which is shown in Figs. 7, 8, 9, 10, 11, and 12 respectively.

## 4 Conclusion

For image classification tasks, ResNet50, a convolutional neural network (CNN) architecture, is especially effective. It makes use of residual connections and enables very deep network training. ResNet50 has achieved great success in image recognition tasks on large-scale images because of its capacity to capture intricate visual patterns and hierarchical representations. A recurrent neural network (RNN) architecture called LSTM is made primarily for processing sequential data. Since of its memory cells and gated architecture, the LSTM is useful in tasks like time series analysis and natural language processing since it can recognize and model temporal dependencies. For the experimental purpose, we took 10 as the epoch and we got the accuracy of ResNet50 as 53% and RNN at 23%. An epoch is the total number of iterations required to train the machine learning model using all of the training data at once. It is measured in cycles. The number of passes a training dataset makes around an algorithm is another way to define an epoch. Based on the experimental result we can conclude that ResNet50 model performed well. Our future research work is extended with different datasets and different models.

## References

1. Attal F, Mohammed S, Dedabrishvili M, Chamroukhi F, Oukhellou L, Amirat Y (2015) Physical human activity recognition using wearable sensors. Sensors 15(12):31314–31338. <https://doi.org/10.3390/s151229858>
2. Shaikh MB, Chai D (2021) RGB-D data-based action recognition: a review. Sensors 21(12):4246. <https://doi.org/10.3390/s21124246>
3. Vaughn A, Biocco P, Liu Y, Anwar M (2018) Activity detection and analysis using smartphone sensors. In: 2018 IEEE international conference on information reuse and integration (IRI). <https://doi.org/10.1109/iri.2018.00022>
4. Khan IU, Afzal S, Lee JW (2022) Human activity recognition via hybrid deep learning based model. Sensors 22(1):323. <https://doi.org/10.3390/s22010323>
5. Ali Mazumder MS, Hossain T, Mehedi Shamrat FM, Jahan N, Tasnim Z, Khater A (2022) Deep learning approaches for diabetic retinopathy detection by image classification. In: 2022 3rd international conference on smart electronics and communication (ICOSEC). <https://doi.org/10.1109/icosec54921.2022.9952159>
6. Sutradhar A, Tajmen S, Dhaly A, Mehedi Shamrat FM., Talukder MS, Khater A (2022) Skin cancer classification and early detection on cell images using multiple convolution neural network architectures. In: 2022 3rd international conference on smart electronics and communication (ICOSEC). <https://doi.org/10.1109/icosec54921.2022.9952115>
7. Hossain T, Jahan N, Mazumder MS, Islam R, Javed Mehedi Shamrat FM, Khater A (2022) COVID-19 detection through deep learning algorithms using chest X-ray images. In: 2022 3rd international conference on smart electronics and communication (ICOSEC). <https://doi.org/10.1109/icosec54921.2022.9951879>
8. Javed Mehedi Shamrat FM, Tasnim Z, Chowdhury TR, Shema R, Uddin MS, Sultana Z (2022) Multiple cascading algorithms to evaluate performance of face detection. Pervas Comput Soc Network 89–102. [https://doi.org/10.1007/978-981-16-5640-8\\_8](https://doi.org/10.1007/978-981-16-5640-8_8)
9. Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L (2021) Review of deep learning: concepts, CNN architectures,

- challenges, applications, future directions. *J Big Data* 8(1). <https://doi.org/10.1186/s40537-021-00444-8>
- 10. Ordóñez F, Roggen D (2016) Deep Convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors* 16(1):115. <https://doi.org/10.3390/s16010115>
  - 11. Singh NK, Suprabhath KS (2021) HAR using Bi-directional LSTM with RNN. In: 2021 international conference on emerging techniques in computational intelligence (ICETCI). <https://doi.org/10.1109/icetci51973.2021.9574073>
  - 12. Goutte C, Gaussier E (2005) A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. *Lect Notes Comput Sci* 345–359. [https://doi.org/10.1007/978-3-540-31865-1\\_25](https://doi.org/10.1007/978-3-540-31865-1_25)
  - 13. Choudhary K, DeCost B, Chen C, Jain A, Tavazza F, Cohn R, Park CW, Choudhary A, Agrawal A, Billinge SJ, Holm E, Ong SP, Wolverton C (2022) Recent advances and applications of deep learning methods in materials science. *npj Comput Mater* 8(1). <https://doi.org/10.1038/s41524-022-00734-6>
  - 14. Human action recognition dataset (nd) Kaggle: your machine learning and data science community. <https://www.kaggle.com/datasets/shashankrapolu/human-action-recognition-dataset>
  - 15. Chen L, Nugent CD, Wang H (2012) A knowledge-driven approach to activity recognition in smart homes. *IEEE Trans Knowl Data Eng* 24(6):961–974. <https://doi.org/10.1109/tkde.2011.51>
  - 16. Zahisham Z, Lee CP, Lim KM (2020) Food recognition with resnet-50. In: 2020 IEEE 2nd international conference on artificial intelligence in engineering and technology (IICAET). <https://doi.org/10.1109/iicaiet49801.2020.9257825>
  - 17. Wang M, Gong X (2020) Metastatic cancer image binary classification based on Resnet model. In: 2020 IEEE 20th international conference on communication technology (ICCT). <https://doi.org/10.1109/icct50939.2020.9295797>
  - 18. Akter S, Shamrat FM, Chakraborty S, Karim A, Azam S (2021) COVID-19 detection using deep learning algorithm on chest X-ray images. *Biology* 10(11):1174. <https://doi.org/10.3390/biology10111174>
  - 19. Shiranthika C, Premakumara N, Chiu H, Samani H, Shyalika C, Yang C (2020) Human activity recognition using CNN & LSTM. In: 2020 5th international conference on information technology research (ICITR). <https://doi.org/10.1109/icitr51448.2020.9310792>

# Using AI Planning to Automate Cloud Infrastructure



Vijay Prakash, Leonardo Freitas, Lalit Garg, and Pardeep Singh

**Abstract** Modern software needs modern infrastructure and architectural design. Cloud architecture solutions reference modern infrastructure, micro-services, and modular infrastructure. Cloud architecture is a reference for scalability, resilience, and also in terms of reachability. Services published on the Internet under the cloud label are accessible to people all around the globe. This paper will produce an Autonomous Agent (AA) software prototype capable of making decisions, giving an initial state and a goal. Many tests and algorithms have been revised during the research to find the correct approach for a suitable software model and abstraction capable of performing these features. Research in the future can focus on using Autonomous Agents as a substitution for the classical approach of cloud operations. The prototype will focus on the AI system using test case scenarios to evaluate the functionality of the decision-making system.

**Keywords** Autonomous agent · Cloud infrastructure · Artificial intelligent · Prototyping model

---

V. Prakash (✉) · L. Garg

Department of Computer Information Systems, University of Malta, Msida, Malta  
e-mail: [vijaysoni200@gmail.com](mailto:vijaysoni200@gmail.com)

L. Garg  
e-mail: [lalit.garg@um.edu.mt](mailto:lalit.garg@um.edu.mt)

V. Prakash  
School of Computer Science, University of Petroleum & Energy Studies, Dehradun, India

L. Freitas  
University of Liverpool, Liverpool, United Kingdom  
e-mail: [lppefreitas@gmail.com](mailto:lppefreitas@gmail.com)

P. Singh  
Department of Computer Science & Engineering, Graphic Era Hill University, Dehradun, India  
e-mail: [pardeep.maan@gmail.com](mailto:pardeep.maan@gmail.com)

## 1 Introduction

Nowadays, it is tough to imagine our lives without technology; computers are used constantly in education, workplaces, and even for leisure and entertainment [1]. With the constant evolution and development of Artificial Intelligence (AI), it is possible to observe many applications for AI in our daily routine [2]. Modern software must be reliable and resilient while maintaining scalability and performance [3]. These attributes significantly increase the software's complexity and increase the risk of hidden bugs on complex lines of code. How to quickly respond to service disruptions [4]? How to respond to failures and accidents fast enough to not affect customers' perceptions [5]? These issues can be addressed with advanced AI techniques and create a system with self-healing characteristics adapting too many situations and reacting instantly after observing service disruption. Modern applications use cloud architectures and platforms extremely focused on scalability and elasticity.

Cloud computing (CC) solutions such as Software-as-a-Service (SaaS) are substituting the classical software engineering approach with standard out-of-the-box scalable structures. CC also gave birth to new study segments; for instance, the infrastructure layer that once was the responsibility of a third-party team can now be part of the software solution and be managed as a piece of software integrating the solution. It is possible due to the Infrastructure-as-Code (IaC) philosophy, which intends to apply software engineering techniques and tools to manage the virtual infrastructure of the software solution as part of the software itself. This approach allows software engineers to create the infrastructure dependencies of its application shipping all the infrastructures simultaneously [6–8].

Combining IaC and AI techniques would result in Autonomous Agents (AAs) capable of deciding which components must be created and making decisions upon cloud operations, such as recreating the application's infrastructure modules. The union of these two areas would result in faster operations reducing the dependency on human interaction for everyday routine tasks on elastic applications in a cloud environment. A system's performance, availability, and resilience must be guaranteed, depending on the Service License Agreement (SLA) [9]. Some systems can never stop, increasing the complexity of their implementation and development. Before all that, the software must be developed, organized, and controlled. In general, these tasks are orchestrated by a formal project management standard, and one of the aspects considered is human relations. AI can be used on many different applications and implementations, even on critical systems; using AI algorithms to automate infrastructure deployment can reduce the time of creation and implementation of those infrastructure elements. A constructive method will combine IaC and AI planning to produce a software prototype capable of making decisions regarding changes perceived in the environment. Standard and modern AI planning algorithms will be researched as well as their applications to find an AI algorithm that would enable the prototype [10, 11]. Standard AI planning algorithms range from pathfinding shortest path (motion algorithms), decision-theoretic or algorithms based on decision theory

or derivatives, and an extension from the previous one but focused on game tree and game theory, generally applied to game software [12].

## 2 Related Work

Many planning algorithms serve propositions, and each algorithm has its own characteristics that fit the purpose of its development. Some planning algorithms are focused on determining a plan for motion and navigation systems for robots that are capable of assembling a car [13] or for navigation systems for autonomous rovers, such as the Mars rovers Curiosity and Discovery [14], which autonomously select the best path to achieve specific coordination, which obstacle to avoid, which terrain is safe to cross, etc. These planning algorithms are broadly classified as Motion Algorithms [15], classical planning such as Stanford Research Institute Problem Solver (STRIPS) [16], Decision-Theoretic algorithms (DTAs) [17], etc.

Decision-Theoretic algorithms seem to work perfectly with reasoning capabilities and imitating human reactions. Due to extending the AI capabilities, it is necessary only to add new actions to the AA arsenal; there is a considerable reduction in the management tasks on the AI code. It is possible to apply this concept by creating a graph network linking the states in an environment by actions in the real world. Assuming the graph network represents the World States, the actions will act as the agents for the transition between these states. Actions have conditions that must be satisfied before the action can be executed. Each action has its effects on the environment, and the result of an action or a sequence of actions can result in the desired World State.

### 2.1 *Cloud Infrastructure*

In cloud computing, a typical pattern gaining notoriety is using software engineering techniques to manage and version infrastructure changes [18]. Once the cloud is the evolution and combination of many virtualization solutions, treating infrastructure and computational resources as a piece of code is not strange. Since the virtual resources are software (compiled software or interpreted code), orchestrating the creation and evolution of the infrastructure as a software code has a lot of benefits. It fits perfectly into the cloud computing domain [19–21]. With infrastructure-as-code (IaC), the infrastructure can be divided into modules corresponding to a layer of the infrastructure solution [22]. Software that needs a database to store its data and an application module to process the user's requests will need servers to run these two components and several network elements to interconnect them. In a cloud environment, the network layer needs to be integrated piece by piece; using Amazon Web Services (AWSs) as an example, the computers would need a Virtual Private Cloud (VPC) with at least two subnets connected to two virtual routers. If the system

needs Internet connectivity, it also needs other elements and the configuration of routes on the virtual routers. Using IaC to manage the infrastructure, the prototype would focus on maintaining infrastructure modules rather than individual pieces.

## 2.2 *Prototype Model*

The prototype will consist of three main parts: A Finite State Machine (FSM), a Planner system (the planner component), and an agent class (AA component), with the AA sensors of the environment. The agent will inspect the cloud environment and search for the virtual components of a software project. Let us consider 3 (three) modules, VPC, DB, and APP. Once the AA sensors detect that the APP module was deleted from the environment, it triggers the planning system to generate an action plan to recreate the APP module. The AA can also detect the application's failures and react by deciding what to do.

## 3 Problem Formulation

With the constant increase in the adoption of solutions using cloud architecture and agile methodologies, it is necessary to evolve the way how environments are provisioned and maintained. Companies that need fast development and deployment are using agile methods to manage their demands. With the rapid pace of agile sprints, it is common to have bottlenecks in the working pipelines, such as the developers waiting for the infrastructure and operations teams to provision a server or an integration.

How much time does it take to create the entire cloud infrastructure of a project? How much time per day does the support team spend inspecting and monitoring the system to prevent service disruption and undesired behaviors? How long does it take for a problem to be identified and solved by the support team? If a problem occurs on the cluster system or the hardware running your software application experiences performance issues or any non-desirable state, what steps are necessary to return the systems to production? How long does a human operator take to react to such situations?

Some of these problems are addressed by implementing project management techniques and using modern and flexible infrastructure components, such as machine virtualization. Continuous integration and continuous delivery are almost sacred terms for solutions being developed for the cloud. IaC addresses the paradigm of dealing with infrastructure elements as software code and using software engineering techniques to manage versioning changes on the infrastructure. Can modern AI techniques be combined with cloud architecture designs on such tasks? Using AI planning techniques would address these problems and drive cloud solutions into the surging era of AI technology.

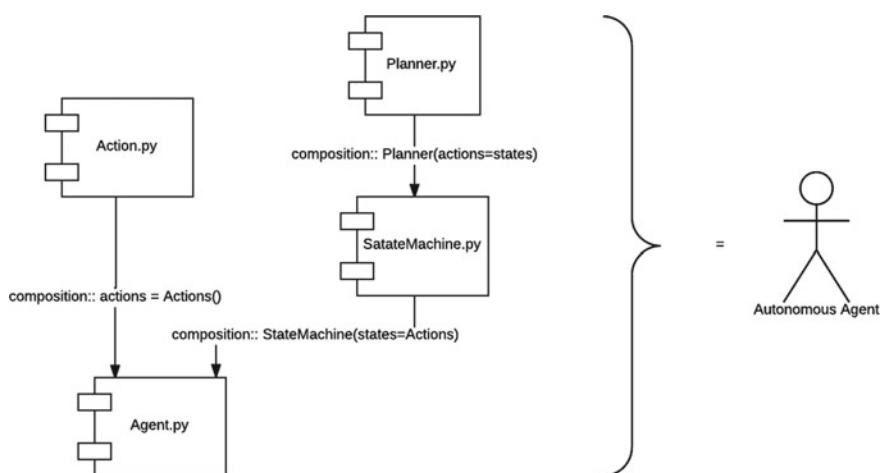
## 4 Proposed Solution

This section covers the design of the prototype, its classes, and the model of abstraction designed for the prototype.

In our prototype, the FSM will transit between states represented as nodes on a graph network, representing a decision tree with pre-categorized states, as depicted in Fig. 1. The AA will be acting as a cloud operator, and its actions and the possible states that the environment can achieve are limited since the scope of the tasks has to obey a sequence. For instance, the AA cannot create a computational instance and deploy an application if the database nor the network (Virtual Private Cloud infrastructure) modules are created since the solution's computational component is directly dependent on these two modules. The agent component of the software is a Python class composed of the classes State Machine and Planner, using the composition as an architecture design solution to avoid multiple-inheritance paradigms.

The planning system would use a searching algorithm to find the best sequence of actions or the shortest sequence by navigating through a graph network in which the nodes are representations of World States and the actions as the edges between the nodes. Using the same approach observed in the Goal-Oriented Action Planning (GOAP) system, the goals will be decoupled from the actions and represented by world facts [23]. The planner takes the list of available actions, calculates the possible states these actions can generate, and transforms it into a graph network. The nodes are the World States, and the edges are the actions that can change the world. The prototype deals with three main infrastructure modules, VPC (networking), database, and application, represented as a HashMap data structure.

The Agent or the AA will use the hierarchical paradigm to orchestrate its behavior. The AA always follows a hierarchical and strict sequence of Sense, Plan, and Act

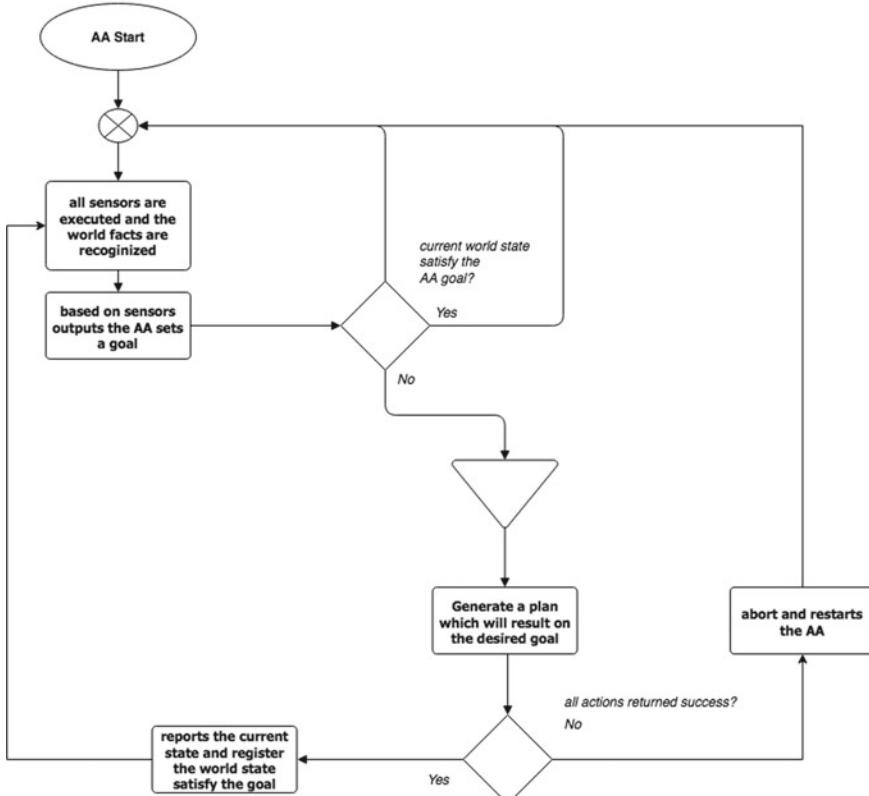


**Fig. 1** Proposed prototype model

steps [24]. On the startup of the AA, it runs a full scan on its sensors to acknowledge the world. Since the AA has a default goal, it checks on the startup if the current state already satisfies the goal; if it is not, the AA executes the planner function to generate an action plan using as starting node the current state as depicted in Fig. 2.

The agent component will be responsible for reading the natural world and transforming its sensor's outputs into valid information for the planner component. The agent class has a collection of sensors constantly measuring the cloud environment. In environmental changes, the sensors set specific goals, so the planner generates an action plan to achieve the goal. Due to the AI planning finding the correct sequence of actions, it needs to traverse the graph network and select the path, which in our model is a sequence of states the AA must transit. To do so, it is necessary to visit the nodes and identify the shortest path traveled; this operation's velocity defines the system's performance and ability to react and plan according to its knowledge actions. The

*AA State Behavior Diagram*



**Fig. 2** Agent state behavior

formulation of the possible states is established with the help of the formulation of the test cases since it will measure the correct abstraction of the represented states.

## 5 Testing and Result Evaluations

To assess the prototype's usability, several test case scenarios were created with the objective of testing possible scenarios that would happen in the real world, creating a simulation of the events perceived by the AA in a cloud environment. The AA must be able to generate an action plan based on any situation exposed to it. The AA must be submitted to situations it can solve with a sequence of actions, searching on its knowledge base for the correct sequence of actions. The User Acceptance Testing (UAT) test cases were elaborated to identify if the AA could deal with scenarios representing actual events that generally occur in cloud applications. For cloud applications, we would consider applications serving web content and web APIs such as RESTful APIs' platforms, stateless web services, and modular web applications in general [25]. Since the developed software is non-interactive, the final user would not deal with streams of commands or configuration files. The prototype is an AA capable of reasoning autonomous tasks based on its decision tree [26]. Based on that fact, the tests' requirements consider assessing the agent's capability for reasoning regarding specific situations. The test cases must consider filling the AA with inputs simulating these situations and the expected output, the product of the reasoning. The test cases are formulated following a template table. The table contains the description of the scenario the test is intended to simulate and the world facts known by the prototype, as depicted in Table 1.

**Table 1** Test case table template

Test case name	Test case identification
Description	Description of the scenario being simulated
Facts	The prototype's sensors recognize the World State <b>World State:</b> { 'vpc': True, 'db': False, 'app': 'out_of_capacity' }
Expected result	The world's expected state must be after the AA executes the action plan. The sensors must perceive the changes performed on the environment and translate them into the world facts it recognizes <b>World State:</b> { 'vpc': True, 'db': True, 'app': True }

**Table 2** Test case 1: empty environment

Test case name	Creating all modules on an empty cloud
Description	In this situation, no infrastructure modules are created in an empty cloud environment. The AA must act and create all infrastructure modules
Facts	In this test case, the AA must identify that the VPC infrastructure module, the database, and the application module do not exist, creating a plan to satisfy this goal <b>World State:</b> {‘vpc’: False, ‘db’: False, ‘app’: False}
Expected result	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: True}

### 5.1 Test Case 1: Empty Environment

In any normal situation in a software project deployment phase, the infrastructure has to be built from the ground. In this test case, the agent must recognize that there is no infrastructure module created on the cloud environment, as tabulated in Table 2.

This test represents a new project being created by the operations team. Still, in this case, the AA will be deployed on the environment and will perform the execution to create the environment from the ground. The AA has planned and executed the plan in less than one second. The plan was created and executed with success.

### 5.2 Test Case 2: No Application Module

In this test case, the application module of infrastructure was deleted, or it was not correctly created and failed on a previous try, as tabulated in Table 3. The AA must create only the application module.

In this scenario, the application module does not exist. In a real-world environment, the application infrastructure module, the instance of the application, is not present and needs to be recreated. The AA has also executed the plan as expected; only the application module has needed to be created, as exposed on the output.

**Table 3** Test case 2: empty environment

Test case name	No application module
Description	In this situation, the application module does not exist
Facts	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: False}
Expected result	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: True}

**Table 4** Test case 3: neither database nor application module

Test case name	Neither the database nor the application module
Description	In this situation, the application module nor the database module exists
Facts	The mock system needs to return to the <b>World State:</b> {‘vpc’: True, ‘db’: False, ‘app’: False}
Expected result	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: True}

### 5.3 *Test Case 3: Neither the Database nor the Application Module*

Both database and application modules do not exist in this scenario, as tabulated in Table 4.

In this scenario, the application module and database do not exist. The AA must detect and create both the database and application modules.

### 5.4 *Test Case 4: Inconsistent Database*

The AA must clean up the environment by destroying the layers and creating new ones. In this situation, the environment might suffer an unexpected issue, such as accidental deletion or compromise of the infrastructure database layer. In this case, the database must be recreated. Due to the extreme scenario of deleting a database, this solution only applies to testing or staging environments. Productive environments need less destructive actions or provisioning scripts to restore the database from the last valid state, as depicted in Table 5.

This is one of the most sensitive scenarios since it involves an inconsistent database and its recreation of it. In this case, the actions performed were as expected; the AA

**Table 5** Test case 4: inconsistent database

Test case name	Inconsistent database
Description	When the AA’s sensors detect an inconsistency in the database For instance, when the database module is not present, the application module should not exist, but it is possible that some disaster has occurred and the database module has gone lost (maybe due to inappropriate human manipulation of the resources). In this situation, the best choice is to clean the environment and generate new resources
Facts	World State: {‘vpc’: True, ‘db’: ‘inconsistent’, ‘app’: True} OR {‘vpc’: False, ‘db’: False, ‘app’: True}
Expected result	<b>World State:</b> {‘vpc’: False, ‘db’: True, ‘app’: ‘stopped’ }

**Table 6** Test case 5: inconsistent database

Test case name	Application module in unhealthy status
Description	The application module exists in this situation, but the computational instance has some issues or may be faulty. In this situation, the app module needs to be stopped
Facts	<b>World State:</b> {‘vpc’: True, ‘db’: False, ‘app’: ‘unhealthy’}
Expected result	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: ‘stopped’}

has identified, destroyed, and recreated the inconsistent database. The app must be stopped for this action since it cannot access the database while it has been deleted.

### 5.5 Test Case 5: Application Module in Unhealthy Status

When the application module is in unhealthy status, as depicted in Table 6.

An application in unhealthy status is an application experiencing issues such as returning HTTP 500 code or any unsatisfactory behavior from the application, and it represents an unstable application. In this case, a more elaborated plan was necessary; the AA has needed to stop the application on unhealthy status, then terminate the stopped app, and then create a new application instance. Note that these tasks could be split since it is a more elaborated plan. Another AA could be responsible for terminating stopped instances once detected, reducing the number of actions one AA performs alone.

### 5.6 Test Case 6: Application Out of Capacity

In this case, the application is suffering capacity issues meaning that more computational capabilities must be created, as depicted in Table 7.

The application is experiencing some performance issues in this scenario, and the sensors have detected it. In this case, the World State shows the app module as ‘out\_of\_capacity’, and the AA needs to increase the application’s capacity by increasing the number of nodes. It can be accomplished in the real world by increasing the number of instances on an Auto Scaling Group or by changing the type of an EC2 Instance on AWS, and it will depend on how the infrastructure was deployed.

The AA had planned as expected, and the result of the action plan was a success. During the test case execution and UAT, assessing the prototype’s usability and the experiment’s possible usage during a real-world implementation was essential. A human operator can learn and execute new procedures to fix an undesired state of the application or environment. This state can be a combination of facts, for

**Table 7** Test case 6: application out of capacity

Test case name	Application out of capacity
Description	In this scenario, the web application being monitored got out of capacity. In a real-world environment, this scenario would be perceived by the AA's sensor as increasing the occurrence of HTTP responses with status code 503. Once the AA's sensor perceives the scenario, it acknowledges the app status as out_of_capacity, making the AA search the sequence of events to restore the World State to its goal where the app status is True
Facts	<b>World State:</b> {‘vpc’: True, ‘db’: False, ‘app’: ‘out_of_capacity’}
Expected result	<b>World State:</b> {‘vpc’: True, ‘db’: True, ‘app’: True}

instance, supposing an application cluster running on AWS, and this application is experiencing performance issues. After the human operator acknowledges the issue, he needs to perform analysis and data information collection to evaluate if there is any procedure associated with that issue and plan which sequence of actions needs to be done to fix the problem; sometimes, the variables are so many that the human operator needs to count on his experience, and some problems need human intuition. The human operator can learn, adapt to unexpected situations, and learn from previous events. The prototype can handle new situations being incremented with new sensors to identify new scenarios and new actions to solve these scenarios. Comparing the AA with the human operator, it must evolve similarly; it is necessary to teach new actions every time new scenarios are mapped and observed.

## 6 Conclusions and Future Scope

The research concluded that it is possible to use planning algorithms to address the automation of task execution in a cloud environment and decision-making processes by giving the AA choices under a decision tree. The limitations of the proposed format are that constant updates must be done to maintain the AA functional in different states and scenarios. Still, it is compatible with the procedures adopted by a human operator since it needs constant training and update of the procedures executed in the environment. Due to the nature of the AA, measuring the impact of some action on a production environment is essential. The prototype would fit perfectly in a development and staging environment since it lacks an SLA for its customers and users, so no incidents would impact the services SLA.

It would be interesting to use libraries allowing the agent to execute parallel tasks, such as listening to the sensors and updating the world facts while the AA executes the transitions; it would permit the AA to be faster and more precise. During the development of the prototype, some alternatives to implement this feature were assessed, but it would drastically change the prototype's architecture and design.

Using a co-routine library and implementing patterns such as an observer listening to events on a channel and executing parallel tasks would be interesting.

Another strategy is to use built-in APIs for low-level programming languages and access them through Python's classes. Languages such as C/C++ and Rust fit this task well since it takes advantage of Python's C-binding API. During the research, an exciting pattern was observed regarding the way the AA would perceive the environment; instead then use a serialized structure in which the AA must observe, plan, and act, and the AA would use reactive behavior reacting upon stimulus on its sensors and not by executing a three steps procedure (observe, plan, and act). This pattern would significantly change the architecture and design of the prototype, so it needs more investigation and planning.

## References

1. Coşkun H (2021) Media as a learning-teaching tool in the context of media literacy from Turkish language teacher candidates' perspective. *Int J Educ Lit Stud* 9(4):93. <https://doi.org/10.7575/aiac.ijels.v.9n.4p.93>
2. van Esch P, Stewart Black J (2021) Artificial intelligence (AI): revolutionizing digital marketing. *Australas Mark J* 29(3):199–203. <https://doi.org/10.1177/18393349211037684>
3. Bannour F, Souihé S, Mellouk A (2022) Scalability and reliability aware SDN controller placement strategies. In: Software-defined networking, pp 41–63. <https://doi.org/10.1002/9781394186181.ch3>
4. Kabadayi S, Tuzovic S, Q Business School. The impact of coronavirus on service ecosystems as service mega-disruptions. Emerald.Com [Online]. Available: <https://www.emerald.com/insight/content/doi/10.1108/JSM-03-2020-0090/full/html>
5. Mack R, Mueller R, Crofts J, Broderick A (2000) Perceptions, corrections and defections: implications for service recovery in the restaurant industry. *Manag Serv Qual Int J* 10(6):339–346. <https://doi.org/10.1108/09604520010352256>
6. Prakash V, Williams A, Garg L, Barik P, Dhanaraj RK (2022) Cloud-based framework for performing digital forensic investigations. *Int J Wirel Inf Netw* 29(4):419–441. <https://doi.org/10.1007/s10776-022-00560-z>
7. Singh P, Prakash V, Bathla G, Singh RK (2022) QoS aware task consolidation approach for maintaining SLA violations in cloud computing. *Comput Electr Eng* 99. <https://doi.org/10.1016/j.compeleceng.2022.107789>
8. Prakash V, Williams A, Garg L, Savaglio C, Bawa S (2021) Cloud and edge computing-based computer forensics: challenges and open problems. *Electronics* 10(11):1229. <https://doi.org/10.3390/electronics10111229>
9. Terry DB, Prabhakaran V, Kotla R, Balakrishnan M, Aguilera MK, Abu-Libdeh H (2013) Consistency-based service level agreements for cloud storage. In: SOSP 2013—Proceedings of 24th ACM symposium on operating system principles, pp 309–324. <https://doi.org/10.1145/2517349.2522731>
10. Chhabra G, Singh P, Yadav H, Bathla G (2023) Industry Internet of Things based intelligent driver monitoring system. In: 2nd International conference on sustainable computing and data communication systems (ICSCDS 2023) Proceedings, pp 1160–1164. <https://doi.org/10.1109/ICSCDS56580.2023.10104883>
11. Singh P, Bathla G, Panwar D, Aggarwal A, Gaba S (2023) Performance evaluation of genetic algorithm and flower pollination algorithm for scheduling tasks in cloud computing, pp 139–154. [https://doi.org/10.1007/978-981-99-1312-1\\_12](https://doi.org/10.1007/978-981-99-1312-1_12)

12. Dhaya BS (2016) Ai-augmented automation for devops, a model-based framework for continuous development in cyber-physical systems [Online]. Available: [www.ijcrt.org](http://www.ijcrt.org)
13. Varlamov O (2021) 'Brains' for robots: application of the Mivar expert systems for implementation of autonomous intelligent robots. *Big Data Res* 25. <https://doi.org/10.1016/j.bdr.2021.100241>
14. Kalita H, Thangavelautham J (2020) Exploration of extreme environments with current and emerging robot systems. *Curr Robot Rep* 1(3):97–104. <https://doi.org/10.1007/s43154-020-00016-3>
15. Oliensis J (2001) A critique of structure-from-motion algorithms. *Comput Vis Image Underst* 84(3):407–408. <https://doi.org/10.1006/cviu.2001.0914>
16. Fikes RE, Nilsson NJ (1971) STRIPS: a new approach to the application of theorem proving to problem solving. *Artif Intell* 2(3–4):189–208
17. Boutilier C, Dean T, Hanks S (1999) Decision-theoretic planning: structural assumptions and computational leverage. *J Artif Intell Res* 11:1–94. <https://doi.org/10.1613/jair.575>
18. Novakouski M, Lewis GA, Anderson WB, Davenport J (2012) Best practices for artifact versioning in service-oriented systems
19. Edmond D, Prakash V, Garg L, Bawa S (2022) Adoption of cloud services in central banks: hindering factors and the recommendations for way forward. *J Cent Bank Theory Pract* 11(2):123–143. <https://doi.org/10.2478/jcbtp-2022-0016>
20. Marinho M, Prakash V, Garg L, Savaglio C, Bawa S (2021) Effective cloud resource utilisation in cloud erp decision-making process for industry 4.0 in the United States. *Electronics* 10(8). <https://doi.org/10.3390/electronics10080959>
21. Njenga K, Garg L, Bhardwaj AK, Prakash V, Bawa S (2019) The cloud computing adoption in higher learning institutions in Kenya: hindering factors and recommendations for the way forward. *Telemat Informatics* 38:225–246. <https://doi.org/10.1016/j.tele.2018.10.007>
22. Achar S (2021) Enterprise SaaS workloads on new-generation infrastructure-as-code (IaC) on multi-cloud platforms. *Glob Discl Econ Bus* 10(2):55–74. <https://doi.org/10.18034/gdeb.v10i2.652>
23. Hartala I (2016) Goal oriented action planning for agent simulations
24. Boeda G (2021) Multi-agent cooperation in games with goal oriented action planner: use case in WONDER prototype project. In: 17th AAAI conference on artificial intelligence and interactive digital entertainment, AIIDE 2021, pp 204–207. <https://doi.org/10.1609/aiide.v17i1.18909>
25. Subramanian H, Raj P (2019) Hands-on RESTful web API design patterns and best practices: design, develop, and deploy highly adaptable, scalable, and secure RESTful web APIs
26. Franklin S, Graesser A (2015) Is it an agent, or just a program? A taxonomy for autonomous agents. In: Lecture notes in computer science (including subseries Lecture notes in artificial intelligence and Lecture notes in bioinformatics), vol 1193, pp 21–35. <https://doi.org/10.1007/bfb0013570>

# Using Historical Trip Information to Determine the Waiting Time Required for Taxi Services



Michael Vassallo, Vijay Prakash, Lalit Garg, and Pardeep Singh

**Abstract** Taxi is a standard mode of transportation in all countries. Whenever a customer requests a taxi service, a waiting time is estimated to indicate how long the client will wait to be picked up. This paper develops prediction models that predict the waiting time for taxi services. These models have been used to develop historical trip information that does not include vehicle GPS coordinates, traffic information, and weather data. Verification has been performed by calculating the accuracy percentage for the classification Random Forest and k-Nearest Neighbours algorithms and the Root Mean Squared Error (RMSE) for the regression Random Forest model. Validation has been done on a new dataset by comparing the predictions with the actual waiting time. The same validation techniques have been used with the waiting time estimated by eCabs. However, the validation results show that the estimated waiting time by eCabs is slightly more accurate than predicted. Integrating GPS vehicle locations, traffic and weather data from a business application would be helpful and should significantly improve the prediction accuracy.

**Keywords** Prediction models · Historical trip information · eCabs · Waiting time

---

M. Vassallo  
University of Liverpool, Liverpool, UK

V. Prakash (✉)  
School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India  
e-mail: [vijaysoni200@gmail.com](mailto:vijaysoni200@gmail.com)

V. Prakash · L. Garg  
Department of Computer Information Systems, University of Malta, Msida, Malta  
e-mail: [lalit.garg@um.edu.mt](mailto:lalit.garg@um.edu.mt)

P. Singh  
Department of Computer Science and Engineering, Graphic Era Hill University, Dehradun, India

## 1 Introduction

The taxi industry is a critical influence on the road efficiency of every country. An effective taxi service improves the commute for the citizens and ensures that the roads are used productively [1]. This paper looks into how a local minicab company can improve its services by automatically predicting the time required for customers to wait from when they request a taxi until they are picked up. The scope of this paper is to predict the waiting time for taxi services using historical trip information without GPS trajectories. This paper aims to use the past three years of trip data to predict the waiting time for new bookings. A literature survey covered existing solutions and theories about feature selection, prediction algorithms, and evaluation metrics. With the acquired knowledge, different feature selection and prediction methodologies were developed, and the best models were evaluated. Further, the main motive is to try and determine whether data with no GPS coordinates can be used to predict the waiting time using machine learning algorithms. The successful development of such a solution will contribute to the world of knowledge for both feature extraction and prediction and the transportation industry. The solution can then be matured and integrated with taxi operational systems [2].

After understanding and preparing the historical trip data, the features to be used in the prediction models were manually filtered based on the experience in the field. Subsequently, a random forest was used further to identify the importance of each feature in the data. According to the sponsor, eCabs provide a waiting time of five minutes to forty minutes. To try and keep the prediction models as simple as possible, the predictor waiting time class was rounded up to the nearest five minute.

Training and test datasets were created from the filtered data and fed into the Random Forest and k-Nearest Neighbours algorithms. Predictions were compared with the actual waiting time using a Confusion Matrix [3] for classification models and RMSE for the regression one. The best-performing algorithms were validated against a new dataset and evaluated by the sponsor. Finally, documentation of all the work, results, and discussions are encompassed in the following sections.

## 2 Related Work

Maltese authorities are trying to promote alternative modes of transportation to solve the transportation problem [4]. With busy schedules and inefficient public transport, people are very attached to their private cars due to the freedom of movement and availability. In the past five years, there has been a significant improvement in the quality of services that the taxi industry offers. With better taxi services and competitive and cheaper prices, people are starting to be more open to travelling with taxis while avoiding the hassle of driving in traffic and finding parking [5]. The pursued literature review concerns existing solutions for problems revolving around feature

extraction and prediction. The aim is to understand how previous studies tackled the problem of extracting the relevant features from the available ones.

Consequently, the focus is on how these features were used to predict a desired outcome accurately. By analysing the approach taken by different studies, the most appropriate algorithms will be used to try and predict the waiting time from this paper's dataset. The main challenge is using these algorithms in a dataset that does not include all factors currently considered when estimating the waiting time. GPS trajectories, vehicle locations, and traffic and weather data are unavailable. Therefore, the main challenge to the hypothesis is whether predicting the waiting time using historical trip information is possible. The authors [6] wanted to try and estimate the current and future state of car traffic in Vermont. In a nutshell, they process data gathered from sensors installed on two main highways, data from online weather systems and other historical data to acquire helpful knowledge and store it in the cloud. Subsequently, they utilise the latter knowledge to predict the traffic in real-time. The algorithm used for extracting features is the CAIM discretisation algorithm. The MIFS, mRMR, and RELIEF algorithms were used to weigh the importance of each feature. Subsequently, Naïve Bayes, k-Nearest Neighbour (k-NN), C4.5, and Random Forest (RF) algorithms were used to predict traffic against each feature extraction algorithm. Tests resulted in RELIEF and k-NN being the most accurate for extracting features and prediction, respectively.

The study by authors [7] is a real-time problem of discovering which stand a taxi should go to after dropping off passengers. The case study is a 441-taxi fleet company operating in Porto, Portugal. According to the authors, at the time of the study, a vehicle waited approximately 44 min to pick up a passenger. As a solution, the authors developed a model that predicts the number of services required at each taxi stand using short, medium, and long-term historical data as input. The authors chose the AutoRegressive Integrated Moving Average (ARIMA) model. Similar to [6], the authors [7] used an algorithm to update the prediction model based on the error measured, which in this case was the Symmetric Mean Percentage Error (sMAPE).

Lam et al. [8] proposed a successful implementation of an European Conference on Machine Learning and Principles and Practice of Knowledge Discovery (ECML/PKDD) competition held in 2015 [9]. The challenge was predicting the final destination and the time for a taxi trip to improve electronic dispatch. The Mean Haversine Distance (MHD) and the Kernel Regression are examples of algorithms used to get the shortest trips and destinations and for matching trips with the same context, respectively. The Random Forest was also used to identify which features can contribute to predicting the final destination. The model was trained using a stacked generalisation approach [10] consisting of the Gradient-Boosted Regression Trees (GBRT) [11], Random Forest regressor [12], and the Extremely Randomised Trees (ERT) regressor [13]. Finally, predictions were evaluated using the Root Mean Squared Logarithmic Error (RMSLE).

A problem studied by Lee et al. [14] is about predicting the airport taxi times in real-time to complement the runway scheduler in airport operations. The authors compare the Linear-Optimised Sequencing (Linos) predictions with those of other machine learning algorithms such as Linear Regression (LR), Support Vector

Machines (SVMs), k-Nearest Neighbours, and Random Forest. In the following problem, the authors [15] conducted, Internet data is utilised to help predict public transport arrivals in areas hosting special events. The authors test different machine learning algorithms against two models, one containing basic features and another with the full features extracted from the Internet and an RFID-based smartcard system. Predictions are then compared using four indicators. Neural Networks, K-Nearest Neighbour, Gaussian processes, radial basis functions, linear regression, regression trees, and support vector regression algorithms from the Weka platform [16] were used to predict arrivals. The correlation coefficient of predicted and actual arrivals, the Mean Absolute Error (MAE), the Root Mean Squared Error of the arrivals, and the Root Mean Squared Error Normalised (RMSN) were used to assess the results of each algorithm. A different study by the authors [17] looks into a model to estimate taxi travel times using the origin and destination GPS trajectories, historical travel time, and distance as training data. They used the Levenberg–Marquardt (LM) method [18] to solve the nonlinear least square problem.

### 3 Problem Formulation

The research hypothesis uses historical trip information to determine the real-time waiting time required for a taxi service. The sponsor for the project is a local (Maltese) minicab company called eCabs. eCabs, founded in 2010, operates 24 h a day, seven days a week. All cabs used to execute taxi services are owned by the company and driven by its employees. The philosophy of eCabs is to revolutionise local transportation through professionalism and technology. They continuously strive to implement industry best practices and intelligent technologies and provide training to their employees. eCabs has successfully reformed the taxi services in Malta by building trust towards cabs by its clientele [19]. Currently, whenever a customer requests an eCabs taxi, an estimated waiting time is provided through the experience and gut feeling of the dispatcher. The latter, responsible for managing the fleet, estimate the waiting time based on the current location of the vehicles, the amount of traffic on the roads, and any other factors that might contribute to the time it takes for a cab to reach the customer.

The problem is that eCabs has reached a threshold in managing the increased number of cabs on the road. Therefore, having an algorithm that automatically and scientifically calculates the waiting time in real-time will relieve the dispatcher from having to worry about all the logistics when providing waiting times. Such a solution would also help the organisation to improve its efficiency and customer trust, thereby keeping up with the organisational goals [20].

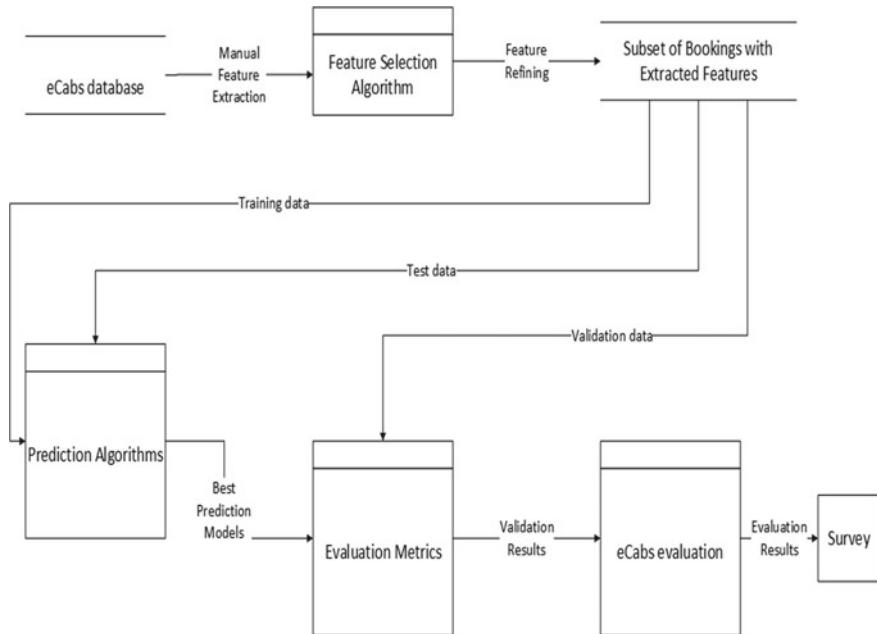
## 4 Proposed Solution: Design and Implementation

The research problem is only to predict the waiting time required for taxi services using historical trip information. Analysing the approach taken for each, one can identify that all predictions were based on features. These features are variables or fields that are part of the data used for prediction and were either chosen manually or through a feature extraction algorithm. Therefore, one of the first things that should be considered as part of the solution to this problem is the required features. Given that experience in the taxi industry and the data available was gathered by having the sponsor, eCabs Company Limited, as a client for several years, the first approach that will be taken to extract the features is by selecting them manually. These features have been discussed and confirmed with the sponsor to verify the selection based on the sponsor's day-to-day experience. Subsequently, feature extraction algorithms have been used to identify the essential features and investigate whether refining the features through these algorithms improves the quality of the predictions.

With the features at hand, the next step would be to use a prediction algorithm to predict the waiting time for taxi bookings. During the literature survey, it was understood that there are no specific rules to define which predictive algorithm to use for each scenario or problem. The way forward would be to try multiple algorithms and identify which returns the most accurate results. In fact, after choosing the algorithms to use for prediction, the evaluation metrics would need to be determined to assess the results based on the test data. Another evaluation metric would be a survey where the dispatchers working at the sponsor's company would rate the solution's effectiveness based on the predicted versus the actual waiting times.

Figure 1 shows how the solution for the research hypothesis will be approached and developed. A snapshot of the eCabs database will be deployed and queried for the relevant bookings. Utilising the experience gathered through a couple of years working with eCabs data. The sponsor has chosen and verified a subset of the most relevant fields to the solution. Subsequently, a feature selection algorithm will refine the feature selection. Having the variable importance at hand, the backward elimination approach will be employed where the least significant features are removed at each run of the prediction algorithms. The best accuracy will be recorded with the features used to achieve it so that the established features are utilised in the prediction algorithms. Data will then be split into training, testing and validation sets.

The training and testing datasets will be used to achieve the best prediction algorithms, whereas the validation dataset to validate the performance of the best prediction algorithm on unseen data. The prediction models implemented are the Random Forest and the k-Nearest Neighbours. Various executions of these models will be done to try and achieve the best prediction results. Each execution must improve the previous one by modifying the arguments required. Since the kNN algorithm ranks the importance of variables by their values, the data will be scaled down to as much as possible on the same scale. Scaling will be done both manually and through a function, where the difference in results will be recorded and observed.



**Fig. 1** Data flow diagram of the system design for the proposed solution

Accuracy and RMSE will be the primary verification and validation statistics used to measure the effectiveness of the prediction models and understand when to stop improving. These statistics will also be computed between the eCabs Waiting Time and the actual Waiting Time so that the results of the statistics will be compared against those of the best prediction algorithms. Comparisons can then be forwarded to the sponsor for evaluation and identify the performance of a prediction algorithm that does not consider traffic, weather, and vehicle locations against that of an eCabs dispatcher who is aware of all factors.

## 5 Results' Evaluation

The section first covers the feature extraction and implementation of the prediction algorithms before looking into the result evaluations. The Random Forest algorithm is used to assess the importance of each feature. A set of 13 relevant features have been extracted for 548,765 bookings, as listed in Table 1.

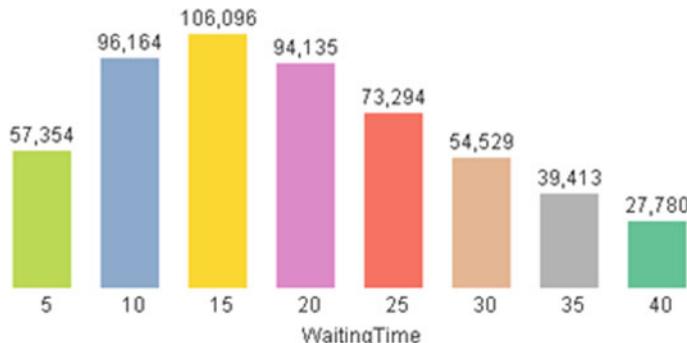
Furthermore, Fig. 2 shows the number of records categorised in each of the eight WaitingTime classes.

The feature's importance returned by the Random Forest method for each variable is shown in Fig. 3.

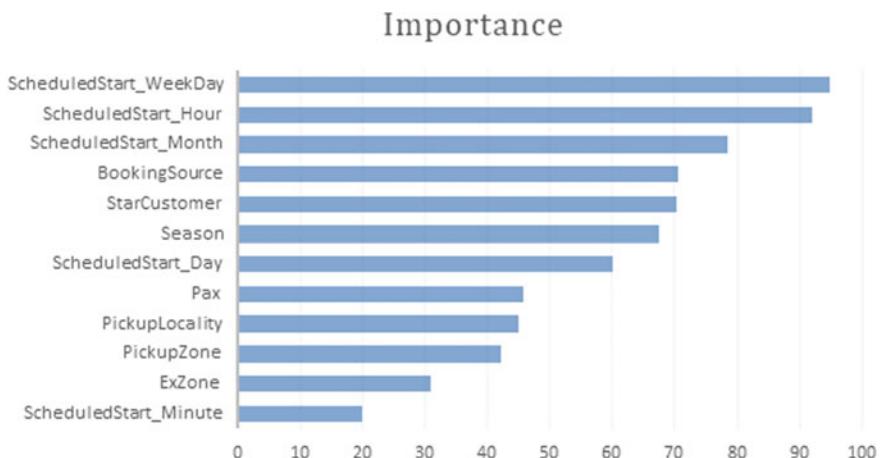
**Table 1** Relevant features for the prediction models

Feature	Description
PickupLocality	Locality for customer pickup
ExZone	Whether customer pickup is in an ExZone, ExZone determines whether the pickup zone is in a famous zone
BookingSource	Source by which booking was created. Values can represent <i>calls</i> , <i>mobile applications</i> , <i>websites</i> , etc.
StarCustomer	Whether the customer is a star customer (important)
PickupZone	Zone for customer pickup
ScheduledStart_Day	The day when booking should start
ScheduledStart_Month	The month when booking should start
ScheduledStart_Hour	The hour when booking should start
ScheduledStart_Minute	The minute when booking should start
ScheduledStart_WeekDay	Field to be created. Day of week when booking should start
Season	The season when booking should start
Pax	Number of passengers
WaitingTime	Class to be used for prediction

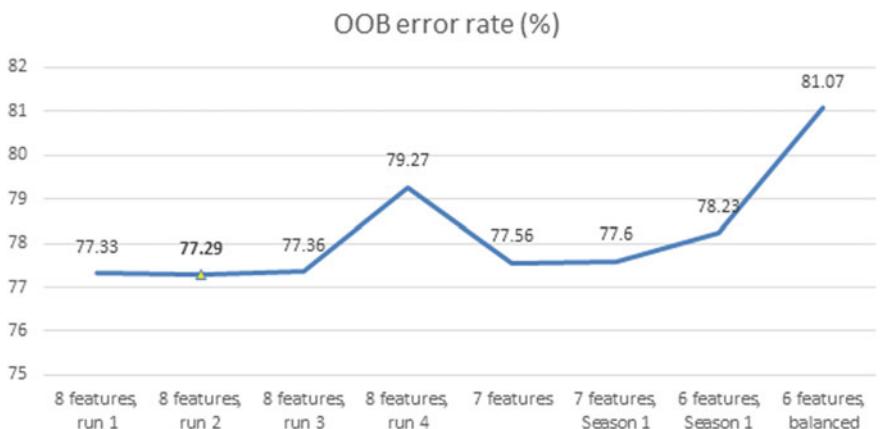
Number of Records by Waiting Time

**Fig. 2** Bar chart showing the number of records by waiting time class

The Random Forest was the first algorithm implemented to try and predict the waiting time. The following algorithm that was executed to try and predict the waiting time is the k-Nearest Neighbours. Using the same approach for the Random Forest model executions, kNN models were run using different features. The *RODBC* library and the *read.csv* function were used to load data retrieved by the normalised query. For each execution of Random Forest, the OOB estimate of the error rate was recorded and is plotted in Fig. 4.



**Fig. 3** Feature importance according to the Random Forest



**Fig. 4** First phase of executions for the Random Forest

This phase of executions started by running a model with the top eight features, omitting the *ScheduledStart\_Minute*, *StarCustomer*, *ExZone*, and *PickupLocality* fields. Another execution with the same number of features was done, specifying that the random forest can use identical records at each split in the trees ( $\text{replace} = T$ ). Since the first run gave a 77.33% error rate and the second run gave a 77.29% error rate, the second model parameters are preferred, and from now on, the  $\text{replace} = T$  will be used when running random forests.

Building upon the second model, the next run was done by omitting the *PickupZone* field instead of the *PickupLocality*. This was done to check whether fewer levels in a field will give better, the same, or worse error rates. The error rate returned

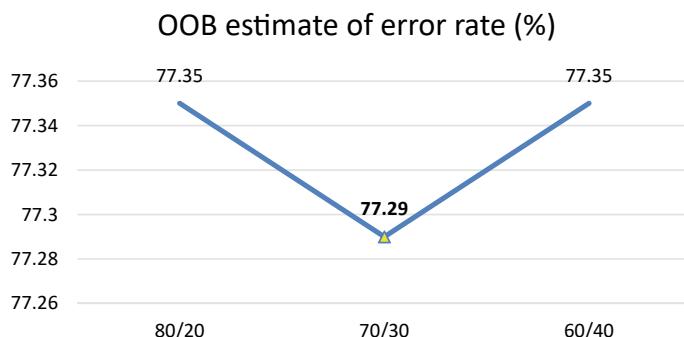
was 77.36%, which is the worst. This shows that fewer levels provide better predictions from the Random Forest model. Finally, the second model was used again with balanced classes of the *WaitingTime* predictor. However, this resulted in a significant increase in the error rate of 79.27%.

The next attempt was to decrease another feature from the dataset. This time, the *season* was removed, achieving an OOB error rate of 77.56%. Another attempt with the same number of features was made, reducing the dataset to bookings that occurred in Season One only. This was done to test whether having a model per season would decrease the OOB error rate. However, the outcome was an error rate of 77.6%, which shows that predictions are less accurate with less data. The last two model executions for this phase utilised six features, removing the *ScheduledStart\_Month* variable. Since there is no seasonality or monthly data, the dataset was again reduced for bookings serviced in Season One. Returning a more significant error rate of 78.23%, the last run was done using all the datasets for six features but having balanced *WaitingTime* classes. This resulted in the worst error rate so far, that of 81.07%.

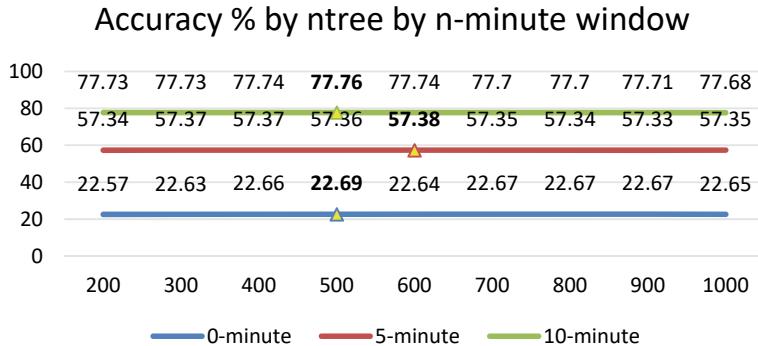
From Fig. 4, one can easily observe that the best prediction classification random forest model achieved so far was when the top eight features were used with the *replace* argument set to *True*. Having the best number of features and knowing which data must be used for the random forest model, the next phase of executions was done by trying different ratios on how the data must be split to train and test the random forest models.

It can be seen from Fig. 5 that both executions have returned the same OOB estimate of error rate (77.35%), which shows that the best ratio to split the training and test datasets remains that of 70% and 30%, respectively.

Nine different random forest executions were done, ranging the number of trees between 200 and 1000. The accuracy for each model was recorded by summing up all the correct predictions done by the model. Additionally, a range of minutes was taken as a window to mimic the definition of accuracy for the sponsor. The different models with the various accuracy windows can be found in Fig. 6.



**Fig. 5** OOB estimate of error rate (%) by different train/test split ratio

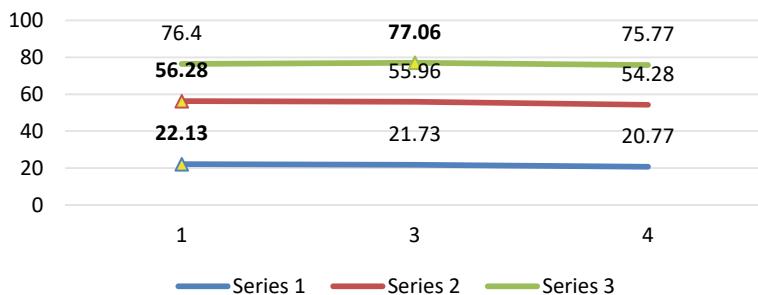


**Fig. 6** Accuracy %age for different values of ntree by various accuracy windows

From Fig. 6, one can observe that the model which predicted the waiting time precisely as it really was with the 0 min window was the one with 500 trees. When a 5 min window is considered, meaning that it is still considered accurate if the predicted waiting time is 5 min away from the real one, the model with 600 trees had the best accuracy. Finally, when we consider accuracy with a 10 min window, the model with 500 trees performed the best again.

Subsequently, another phase for the random forest model was to try different values for *mtry*. Since the sponsor specified that a 5 min window could be used to consider the prediction model accurate, the prediction model with 600 trees was used in this execution phase. Three models were run, with 1, 3, and 4 variables considered at each split. The accuracy of these models can be seen in Fig. 7.

Figure 7 shows that the best classification method for the 0 and 5 min window is with *mtry* equal to one. The best model for the 10 min window was with *mtry* equal to 3. Having said that, since the previous executions returned a better accuracy, these models will be discarded. The last execution was done by running the best classification Random Forest model on one year of data. As can be seen in Fig. 8, the accuracy percentage for the 0 min window is 22.48%, the 5 min window is 56.99%, and the 10 min window is 77.6%. Analysing the accuracy percentages returned by

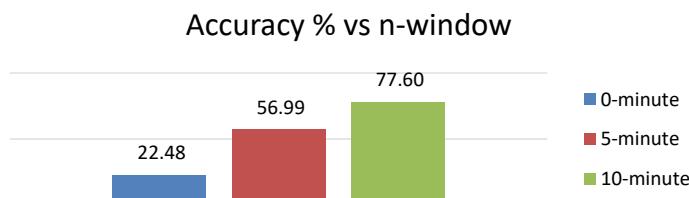


**Fig. 7** Chart showing accuracy %age by different mtry

all the executed random forest models, one can conclude that the best accuracy was achieved using the arguments in Table 2.

Utilising the same number of features and keeping the *replace* argument set to *True*, different random forest models were executed without converting the *Waiting-Time* class to a factor. Since a regression model was implemented in each run, the verification was done by recording the MSE and RMSE and plotting them in Fig. 9.

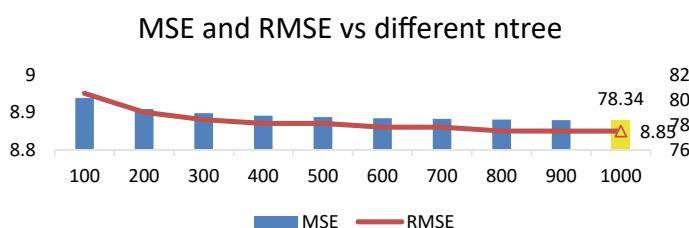
It can be seen from Fig. 9 that the best model with the least error was when 1000 trees were used for the forest. Another execution done upon the last best model was using only one year of data. Interestingly enough, the MSE decreased to 78.29 and the RMSE to 8.85. This means that the model with one year of data returned the best predictions for the random regression forest. Following the same approach for the



**Fig. 8** Accuracy %age versus n-window using the best classification Random Forest for one year of data

**Table 2** Properties for Random Forest that resulted in the best accuracy

Property	0 min window	5 min window	10 min window
Number of features	8	8	8
Training/test ratio	70/30	70/30	70/30
Number of trees (ntree)	500	600	500
Number of features (mtry)	2	2	2
Replace	True	True	True
Full dataset?	Yes	Yes	Yes
Number of features	8	8	8



**Fig. 9** MSE and RMSE against different regression models with different ntree

Random Forest, the most accurate kNN models were achieved using the parameters in Table 3.

Figure 10 compares the accuracy percentage of the best classification random forest model and the kNN model. Although there are no significant differences, it is evident that the Random Forest model outperformed the kNN model in all n-window accuracy scenarios. The properties used for both models are summarised in Tables 2 and 3.

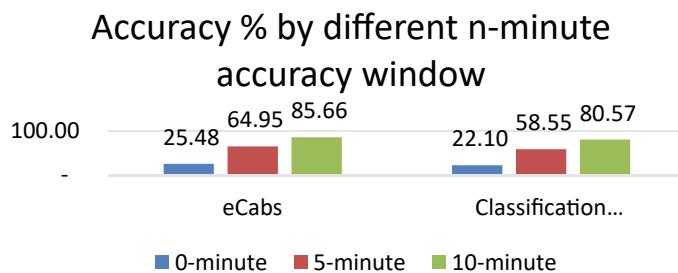
Furthermore, to compare the latter accuracy percentage and RMSE with the current method for estimating the waiting time, the same calculations were done against the eCabs estimates. The comparison results for the accuracy % age are in Fig. 11, and those for the RMSE are in Fig. 12.

**Table 3** Properties for kNN model that resulted in the best accuracy

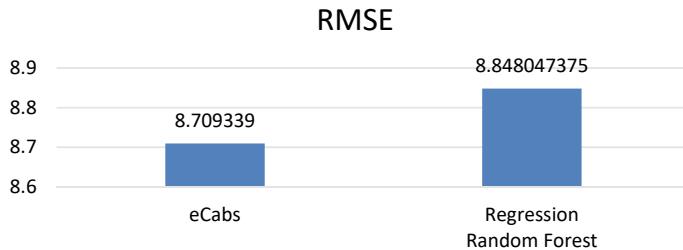
Property	0 min window	5 min window	10 min window
Number of features	8	8	8
Training/test ratio (%)	80/20	80/20	80/20
Number of neighbours (k)	431	87	67
Normalised?	False	False	False
Weighted?	False	False	False
Full dataset?	Yes	Yes	Yes
Number of features	8	8	8



**Fig. 10** Accuracy % by n-minute accuracy window by the best prediction models



**Fig. 11** Accuracy % age comparing the eCabs estimated values versus the predicted values



**Fig. 12** RMSE comparing the eCabs estimated values versus the predicted values

## 6 Conclusions and Future Scope

In this paper, Random Forest and k-Nearest Neighbours algorithms models predicted the waiting time for eCabs taxi services. Verification has been performed by calculating the accuracy percentage for the classification Random Forest and k-Nearest Neighbours algorithms and the Root Mean Squared Error for the regression Random Forest model. Validation has been done on a new dataset by comparing the predictions with the actual waiting time. According to the sponsor, the current waiting time estimates are not so accurate. One reason would be that even though the accuracy is a bit less than that of eCabs, being automated would introduce the possibility of handling the ever-increasing number of bookings without the potential problems humans face. However, eCabs operations would improve this solution by integrating it into their operational system, including additional data such as traffic and weather. Additionally, in the future, the research hypothesis can be expanded to include extra information other than historical trip information only, to understand how extra data affects the prediction accuracy.

## References

1. Porter L et al (2018) The autonomous vehicle revolution: implications for planning/the driverless city?/autonomous vehicles—a planner's response/autonomous vehicles: opportunities, challenges and the need for government action/three signs autonomous vehicles will not lead to. *Plan Theory Pract* 19(5):753–778. <https://doi.org/10.1080/14649357.2018.1537599>
2. Vassallo K, Garg L, Prakash V, Ramesh K (2019) Contemporary technologies and methods for cross-platform application development. *J Comput Theor Nanosci* 16(9):3854–3859. <https://doi.org/10.1166/jctn.2019.8261>
3. Amalia R, Wibowo A (2020) Prediction of the waiting time period for getting a job using the Naive Bayes algorithm. *Res J Adv Eng* 5(2):225–229 [Online]. Available: <http://irjaes.com/wp-content/uploads/2020/10/IRJAES-V5N2P219Y20.pdf>
4. Bajada T, Titheridge H (2017) The attitudes of tourists towards a bus service: implications for policy from a Maltese case study. *Transp Res Procedia* 25:4110–4129. <https://doi.org/10.1016/j.trpro.2017.05.342>

5. Brown A, LaValle W (2021) Hailing a change: comparing taxi and ridehail service quality in Los Angeles. *Transportation (Amst)* 48(2):1007–1031. <https://doi.org/10.1007/s11116-020-10086-z>
6. Li H, Li Z, White RT, Wu X (2012) A real-time transportation prediction system. In: Lecture notes in computer science (including subseries: Lecture notes in artificial intelligence and Lecture notes in Bioinformatics), vol 7345, pp 68–77. [https://doi.org/10.1007/978-3-642-31087-4\\_8](https://doi.org/10.1007/978-3-642-31087-4_8)
7. Moreira-Matias L, Gama J, Ferreira M, Mendes-Moreira J, Damas L (2013) Predicting taxi-passenger demand using streaming data. *IEEE Trans Intell Transp Syst* 14(3):1393–1402. <https://doi.org/10.1109/TITS.2013.2262376>
8. Lam HT, Diaz-Aviles E, Pascale A, Gkoufas Y, Chen B (2015) Blue taxi destination and trip time prediction from partial trajectories. In: CEUR workshop proceedings, vol 1526
9. Martínez-Usó NA, Mendes-Moreira J, Moreira-Matias L, Kull M, Lachiche (2015) ECML-PKDD-DCs 2015
10. Martin E et al (2011) Stacked generalization. *Encycl Mach Learn* 912. [https://doi.org/10.1007/978-0-387-30164-8\\_778](https://doi.org/10.1007/978-0-387-30164-8_778)
11. Friedman JH (2008) Greedy function approximation : a gradient boosting machine. *Inst Math Stat* 29(5):1189–1232 [Online]. Available: <https://statweb.stanford.edu/~jhf/ftp/trebst.pdf>
12. Habe H (2012) Random forests, ランダムフォレスト . Inf Process Soc Japan (IPSJ) SIG Tech Rep 情報処理学会研究報告, 2012-CVIM(31):1–8
13. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. *Mach Learn* 63(1):3–42. <https://doi.org/10.1007/s10994-006-6226-1>
14. Lee H, Malik W, Zhang B, Nagarajan B, Jung YC (2015) Taxi time prediction at Charlotte airport using fast-time simulation and machine learning techniques. <https://doi.org/10.2514/6.2015-2272>
15. Pereira FC, Rodrigues F, Ben-Akiva M (2015) Using data from the web to predict public transport arrivals under special events scenarios. *J Intell Transp Syst Technol Plan Oper* 19(3):273–288. <https://doi.org/10.1080/15472450.2013.868284>
16. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software: an update. *ACM SIGKDD Explor Newslett* 11(1):10–18
17. Zhan X, Hasan S, Ukkusuri SV, Kamga C (2013) Urban link travel time estimation using large-scale taxi data with partial information. *Transp Res Part C Emerg Technol* 33:37–49. <https://doi.org/10.1016/j.trc.2013.04.001>
18. Zhang C, Ordóñez R (2012) Numerical optimization. In: Advances in industrial control, no 9781447122234, pp 31–45
19. eCabs—Malta’s Best TaxiApp—Malta eCabs. <https://www.ecabs.com.mt/>. Accessed 02 Feb 2022
20. SurveyMonkey.com. Create the best surveys. <https://www.surveymonkey.com/>. Accessed 01 Mar 2022

# The Impact of Cesarean Section Trends and Associated Complications in the Current World: A Comprehensive Analysis Using Machine Learning Techniques



K. Mallikharjuna Rao , Harleen Kaur, and Sanjam Kaur Bedi

**Abstract** The accurate prediction of the correct mode of delivery is crucial for the safety and well-being of both mother and child. Currently, this decision heavily relies on the subjective judgement of the attending physician, which can introduce risks if an incorrect method is chosen. Many expectant mothers may opt for a cesarean section without fully understanding whether it is the most suitable option for them. Particularly in developing countries, complications during delivery pose significant challenges. This study aims to address these concerns by identifying key features for determining the delivery mode and applying various machine learning algorithms to predict it accurately. The analysis involved five machine learning models, namely K-nearest neighbours (KNN), random forest, decision tree, support vector machine (SVM), and AdaBoost. The dataset utilized in this study consists of 6157 birth records from four different hospitals in Spain, encompassing 161 distinct features. By leveraging regression analysis-based machine learning methods, we strive to enhance the decision-making process and ultimately improve the safety outcomes for mothers and infants.

**Keywords** Childbirth · Vaginal · Delivery · Supervised machine learning · Ensemble methods · AdaBoost · Random forest · Decision tree · SVM · KNN · SMOTE

## 1 Introduction

The choice of delivery mode stands as a pivotal concern for gynaecologists, health authorities, and expectant mothers today. Towards the end of pregnancy, the baby can be delivered either through vaginal delivery, a traditional non-invasive technique,

---

K. Mallikharjuna Rao  · H. Kaur · S. K. Bedi  
Data Science and Artificial Intelligence, International Institute of Information Technology Naya Raipur, Raipur, India  
e-mail: [mallikharjuna@iiitnr.edu.in](mailto:mallikharjuna@iiitnr.edu.in)

or through cesarean section, a contemporary surgical method [1]. Vaginal delivery remains the preferred approach due to its lower rates of morbidity and mortality compared with cesarean section. In the latter, the baby is birthed via an incision made in the mother's uterus [2]. Both delivery methods possess their own merits and drawbacks and selecting the appropriate method for an individual woman necessitates careful consideration of her unique characteristics. Unfortunately, in many developing countries like Bangladesh, improper delivery method selection can lead to temporary or permanent complications for women, encompassing risks such as fatal abortion, internal bleeding, and, in some instances, respiratory issues for the new-born.

Cesarean section, commonly employed in cases of delivery complications, serves as a crucial method to safeguard the health of both the baby and the mother. It is a preferred approach when a normal vaginal birth poses risks to either party. Cesarean section proves beneficial in situations where labour is absent or progressing slowly when the baby is unusually large, positioned in a breech presentation, or experiencing inadequate oxygen supply [3]. Additionally, the mother's health considerations significantly influence the choice of delivery mode. For instance, if the mother is infected with conditions like HIV or genital herpes, a cesarean section is favoured to prevent the transmission of the infection to the baby [3]. Likewise, in cases where the mother suffers from high blood pressure or diabetes, standard vaginal delivery may pose difficulties, making a cesarean section the preferred option. Opting for standard delivery in such circumstances can lead to organ damage and other complications.

Against the backdrop of ongoing advancements across various fields, artificial intelligence (AI) has made significant strides, revolutionizing numerous sectors. Within this context, the cesarean section has emerged as a widely acknowledged modern technology offering a potentially less painful alternative. However, many countries with underdeveloped medical sectors are promoting cesarean sections without fully comprehending the complications it may pose for both mothers and infants. Notably, in Bangladesh, the incidence of avoidable cesarean sections increased by 51% in 2016–2017, with 77% of cases in 2018 being medically unnecessary. Furthermore, the maternal mortality ratio in Bangladesh in 2017 stood at 173 per 100,000 live births, surpassing rates in most developed countries. By comparison, the United States recorded 17 deaths per 100,000 live births during the same year [4]. Additionally, the cesarean section rate in Bangladesh in 2015, as per the Institute of Public Health Nutrition's National Low Birth Weight Survey, reached 35%, surpassing the World Health Organization's recommended range of 10–15%. Consequently, maternal mortality and morbidity rates are approximately twice as high in cesarean section procedures as in standard delivery.

Typically, the attending physician holds the authority to determine the appropriate delivery mode for each case. However, the integration of obstetric apps has proven invaluable in aiding doctors predict the optimal mode of delivery and mitigate complications during childbirth. Extensive research has been devoted to forecasting pregnancy outcomes, leading to the development of clinical decision support systems. Among these advancements, personalized prediction tools have emerged, specifically targeting normal birth after cesarean section using a range of machine learning (ML)

algorithms. Notably, data mining classification models have been devised to enable real-time prediction of delivery modes based on factors associated with birth risks. These efforts collectively contribute to enhancing the accuracy and efficiency of delivery mode decision-making, offering valuable support to healthcare providers in delivering optimal care to expectant mothers.

The subsequent sections of this paper delve into additional studies and findings related to the ongoing research. However, it is essential to highlight that further investigations are warranted to ascertain the suitability of specific delivery modes based on an individual's unique characteristics. Moreover, identifying the most accurate algorithms for predicting a mother's probability of undergoing a cesarean section remains an area requiring further exploration [5]. Hence, this research endeavour was undertaken with distinct objectives in mind.

### 1. Emphasizing Key Features for Delivery Mode Prediction

Our primary focus is to prioritize the essential features that play a crucial role in accurately predicting the mode of delivery. By identifying and analyzing these significant factors, we aim to enhance the precision and reliability of delivery mode predictions.

### 2. Exploring Machine Learning Algorithms for Optimal Predictive Accuracy

We aim to investigate various machine learning algorithms to determine which model yields the highest level of predictive accuracy. By assessing and comparing the performance of different algorithms, we seek to identify the most effective approach for predicting the likelihood of a cesarean section. This analysis will contribute to refining and optimizing the prediction models employed in the field of obstetrics.

The subsequent sections of this paper are structured as follows: Sect. 2 presents a comprehensive summary of existing research, theories, and findings related to the prediction of delivery modes. Section 3 elaborates on the techniques, models, and approaches utilized to prioritize important features and predict delivery modes accurately. The accuracy rates, performance evaluations, and comparative analyses of the models used for delivery mode prediction are thoroughly discussed in Sect. 4. In the final Sect. 5, a comprehensive discussion is conducted, incorporating the implications and significance of the research findings. The conclusions drawn from the study, along with their potential impact on clinical practice and future research directions, are also highlighted.

## 2 Literature Review

Alamet et al. [1] conducted a study specifically investigating the performance of bagging ensemble classifiers in predicting births. The study unequivocally concluded that the bagging ensemble classifiers exhibited superior performance compared with traditional machine learning algorithms for this task. In a separate study by Khan et al. [6], supervised ensemble machine learning models, namely AdaBoost, Cat Boost, and XGBoost were employed. However, the results of this study did not yield

remarkable outcomes, possibly due to the limited consideration of only 11 features for predicting delivery.

Conducting a comprehensive and systematic literature review was an integral part of this research. Islam et al. [4] conducted an insightful study focused on identifying the most effective machine learning algorithms for predicting delivery modes. Their investigation involved the utilization of decision trees, random forest, K-nearest neighbours (KNN), and support vector machines (SVM) algorithms to achieve accurate predictions. Notably, the accuracy rates of these models varied between 74 and 97% when applied to diverse sets of features, ranked based on their importance. The findings of their study shed light on the effectiveness of these predictive models in determining the appropriate mode of delivery. Additionally, the study categorized the factors influencing delivery mode into distinct categories, ranging from less important to highly significant. However, it is important to note that comprehending the research findings may require a substantial effort, as the subject matter delves into various types of deliveries and employs terminologies that are more familiar to medical professionals than the general population.

Campillo-Artero et al. [7] conducted a study aimed at exploring the factors contributing to emergency cesarean section using machine learning classification algorithms, namely logistic regression, classification tree, and random forest. The accuracy rates achieved by these models in the study ranged from 74 to 81%. The focus of this investigation was on a specific subgroup of mothers, specifically those aged 35 and older, who exhibited a higher body mass index (BMI) and had a higher likelihood of having undergone a previous cesarean section. The study aimed to examine how these factors influenced the occurrence of emergency cesarean sections, providing valuable insights into the predictive capabilities of the employed machine learning models.

In a study conducted by Abbas et al. [5], explored the correlation between the probability of having a cesarean section and the age of the mother. The findings of the study revealed that mothers who were younger than 20 years old or older than 35 years old had a higher likelihood of undergoing a cesarean section. Additionally, the study concluded that women who had a cesarean section tended to have higher blood pressure compared with women who typically give birth [4]. This research provides important insights into the association between maternal age, mode of delivery, and blood pressure levels, contributing to our understanding of the factors influencing cesarean section rates.

Wie et al. [8] undertook a precise study to predict emergency delivery using machine learning algorithms. Among the models evaluated, logistic regression demonstrated the highest performance. The study's findings established a significant association between maternal weight before delivery and the likelihood of undergoing a cesarean section. However, maternal weight at delivery did not contribute to the prediction. Furthermore, the study identified a heightened risk of cesarean section with increasing birth weight of the new-borns. Notably, the study's limitations primarily stemmed from its retrospective cohort design.

Jamboo [9] investigated various models and concluded that the Naïve Bayesian approach exhibited superior performance compared with other models tested. The

study's results were deemed realistic due to the use of a cross-validation approach to evaluate the classifier's performance.

In conclusion, the literature review identified several issues in the existing research. Various studies explored the prediction of cesarean sections using different traditional machine learning algorithms. Additionally, some studies investigated the correlation between various features of a woman and the likelihood of a cesarean section, while others aimed to identify the most significant factors in determining the type of birth. Based on this comprehensive review, this study aims to enhance accuracy by focusing on predicting cesarean sections in pregnant women using specific machine learning algorithms and ensemble techniques.

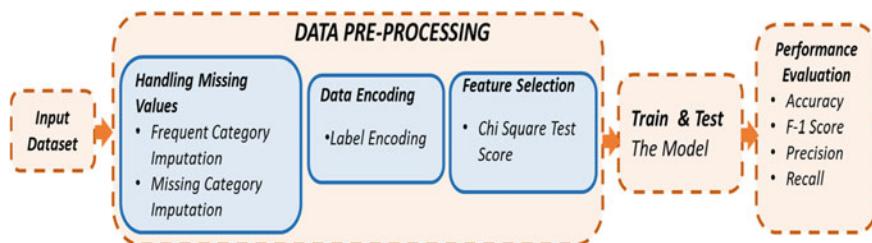
To summarize, the literature review highlighted several issues in the existing research. Numerous studies have concentrated on predicting cesarean sections using various traditional machine learning algorithms. Simultaneously, some investigations aimed to establish relationships between different features of a woman and the likelihood of a cesarean section based on these features, while others sought to identify the most crucial features for determining the type of birth. Building upon the insights gained from the literature review, this study specifically focuses on predicting cesarean sections in pregnant women. It employs specific machine learning algorithms and ensemble machine learning techniques to enhance the accuracy of predictions.

### 3 Proposed Methodology

The research work to build the proposed model was conducted in four phases, namely data collection, data preprocessing, training of the model, and performance analysis of the model. The above phases are briefly described in the following subsections. The pictorial representation of the proposed model is shown in Fig. 1.

#### Phase 1: Data Collection (Input Dataset)

The dataset used in this study was created by Campillo-Artero et al. [7] to develop effective and efficient machine learning models. It comprises birth records from four



**Fig. 1** Proposed model

public hospitals in Spain, specifically from the year 2014. The dataset consists of 6157 single birth records and includes a total of 161 features. Among these features, there are 141 categorical, 19 numerical, and one unnamed feature. Some columns in the dataset were unnamed and therefore removed. The dataset encompasses various pre and postnatal attributes that are valuable for predicting different aspects related to maternal and child health, such as amniotic fluid, oxytocin levels, previous cesarean section, age, height, weight, intrapartum foetal pH, mode of delivery, and more.

## Phase 2: Data Preprocessing

In this stage, the collected data underwent several preprocessing steps. Firstly, the data were merged, and any duplicate entries and noisy data were eliminated from the dataset. To handle missing values, various imputation techniques were applied, such as mean and median imputation for numerical features and filling missing values with frequent values or a designated category for categorical features. Since machine learning models require numerical data, categorical coding was performed. Specifically, label coding was implemented, which transformed the categorical variables into a numerical representation. In alignment with the study's objective of predicting cesarean section, irrelevant features were removed from the dataset. From the remaining set of features, a subset of 32 important features, as identified in the reference work [4], was selected for further analysis and modelling. These selected features were considered relevant and informative for the proposed model.

## Phase 3: Training the Model

In this phase, a set of widely recognized machine learning algorithms, including decision tree (DT), K-nearest neighbour (KNN), and support vector machine (SVM) were chosen to train the model. Additionally, ensemble machine learning techniques such as AdaBoost and random forest were employed to classify the subject, specifically the type of birth, as either cesarean or non-cesarean. Ensemble methods are supervised ML techniques that combine predictions from multiple base models to create an optimized prediction model. This approach leads to improved predictive performance compared with using a single model. The implementation of these models was carried out using Scikit-learn, a Python module that encompasses numerous machine learning algorithms [1].

### 3.1 Decision Tree (DT)

The decision tree is a supervised machine learning technique that is valuable for both classification and regression tasks. In this particular study, the decision tree is employed for classification purposes. It constructs a prediction model by representing the features as a tree structure. The predictions made by the decision tree are derived from splits based on the features. A decision tree comprises root and leaf nodes. The root nodes determine which feature should be chosen for splitting the tree. They aid in assessing the feature that provides the most effective division of the data. On the



**Fig. 2** Flow diagram of a decision tree classifier

other hand, the leaf nodes represent the final nodes of the tree. The selection of root nodes, or the crucial nodes, can be performed using diverse methods, such as features with high entropy, features with the most significant information gain, or the Gini index [10].

The decision tree algorithm follows the following steps:

1. Input the dataset.
2. Define the hyperparameters, such as the criterion for splitting (entropy or Gini index).
3. Select the root node and partition the data based on the chosen criterion.
4. Repeat the splitting process recursively until reaching a stopping condition, typically, when a leaf node is reached.
5. Train the decision tree model using the training dataset.
6. Test the trained model on unseen data to evaluate its performance.

Figure 2 provides a visual representation of the decision tree's block diagram, illustrating the steps involved in constructing and using the decision tree model.

### 3.2 Random Forest (RF)

Random forest is an ensemble machine learning technique that leverages bagging methods. This approach involves dividing the complete dataset into several randomly generated subsets, with each subset used to train a base model independently and in parallel. Random forest is a versatile method applicable to both classification and regression tasks. It derives predictions by aggregating the predictions of multiple base models. The model is generated by constructing decision trees from the data subsets, and the final output is determined based on the average or majority score.

In random forest, the decision tree serves as the foundational base model. It randomly selects observations from the dataset and constructs a decision tree based on these selected samples. The majority result among the individual trees is employed for classification purposes. Each tree independently predicts the class, and the majority class across all trees is adopted as the final prediction. However, due to the generation of multiple trees, random forest is relatively slower compared with a single decision tree [11].

Figure 3 illustrates the process of random forest, highlighting the division of data into subsets, the parallel execution of base models, and the final prediction based on averaging or majority voting.



**Fig. 3** Flow diagram of random forest classifier

The steps involved in the random forest algorithm are as follows:

1. Input the dataset.
2. Set the hyperparameters, including the criterion (such as entropy or Gini) and the number of samples.
3. Construct a decision tree for each sample by randomly selecting subsets of the data.
4. Train each decision tree and obtain predictions for the target variable.
5. Perform voting for each predicted outcome and consider the majority-selected prediction as the final outcome.

### 3.3 K-Nearest Neighbour (KNN)

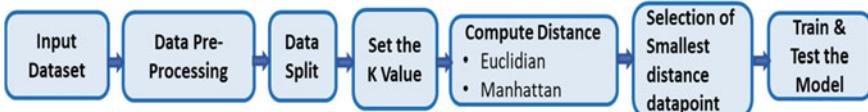
K-nearest neighbours (KNN) is a nonparametric lazy-learner algorithm that does not make explicit assumptions about the data distribution. It requires less training work and more prediction work compared with other algorithms. KNN operates under the assumption of similarity between a new data point and the available cases, assigning the new data point to the category that is most similar to the existing categories. This algorithm can be used for both classification and regression tasks.

KNN calculates the distance between the new data point and the neighbouring points, selecting the shortest distances. Various distance measures, such as Euclidean distance, Manhattan distance, and Chebyshev distance can be used to determine the distances between two data points. Due to its reliance on distance calculations, KNN is often referred to as a distance-based algorithm [12].

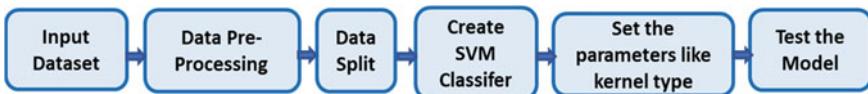
The K-nearest neighbours (KNN) algorithm follows the following steps:

1. Input the dataset.
2. Set the hyperparameters, such as the number of neighbours ( $k$ ) and the distance metric to be used.
3. Calculate the distance between the new data point and the other data points using distance measures like Euclidean, Manhattan, and Chebyshev.
4. Sort the distances in ascending order and select the first  $K$  entries.
5. Retrieve the labels of the selected entries and determine the mode (most frequent label) among them.
6. Train and test the model using the selected  $K$  neighbours.

The block diagram illustrating the KNN algorithm can be seen in Fig. 4.



**Fig. 4** Flow diagram of KNN classifier



**Fig. 5** Flow diagram of SVM classifier

### 3.4 Support Vector Machine (SVM)

The SVM algorithm stands out from other machine learning algorithms due to its ability to efficiently handle multiple continuous and categorical variables. It can be utilized for both regression and classification tasks. In classification, the primary objective of SVM is to determine an optimal line or decision boundary that effectively divides the feature space into distinct classes. This optimal decision boundary is referred to as a hyperplane [4]. The SVM technique seeks to identify a hyperplane that maximizes the margin between two classes. In cases where class separation is not straightforward, a kernel function can be employed to transform the hyperplane's dimensions from 1 to n.

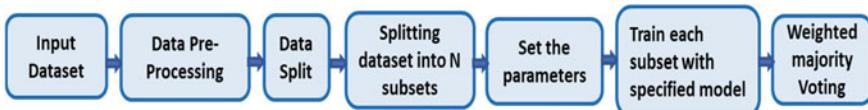
Steps in SVM technique:

1. Load the dataset.
2. Choose a kernel function, such as RBF, linear, or polynomial.
3. Train the SVM model using the selected kernel function.
4. Make predictions and test the trained model.

Refer to Fig. 5 for the block diagram representation of SVM.

### 3.5 AdaBoost (AB)

AdaBoost is a machine learning technique that improves the performance of classifiers by utilizing an ensemble of decision trees. Its purpose is to construct a robust predictive model by iteratively learning from the mistakes made by weak classifier models. During each iteration, misclassified data points are identified and assigned higher weights, while correctly classified points are assigned lower weights. This weighting scheme ensures that subsequent classifiers focus more on the misclassified points and aim to correct them. In AdaBoost, the training data is sampled from an updated distribution, and the classifiers are combined using weighted majority



**Fig. 6** Flow diagram of AdaBoost method

voting. This iterative process enhances the overall accuracy and effectiveness of the ensemble model [4].

The steps in the AdaBoost technique are as follows:

1. Load the dataset.
2. Specify hyperparameters, including a base estimator, number of estimators, learning rate, and random state.
3. Build the initial model and make predictions.
4. Update the weights, assigning higher weights to misclassified points and lower weights to correctly classified points.
5. Build the next model using the updated weights and make predictions.
6. Repeat steps 4 and 5 for the specified number of estimators.
7. Combine the models using weighted averaging to create the final model.
8. Train and test the model.

Refer to Fig. 6 for the block diagram representation of AdaBoost.

#### Phase IV: Performance Evaluation

The cross-validation technique is a statistical method used to assess the performance of machine learning models. In this study, the stratified K-fold cross-validation technique is employed, specifically suitable for unbalanced datasets. This technique divides the entire dataset into k folds, ensuring an equal distribution of instances with and without cesarean section in each fold.

**F<sub>1</sub> Score**—F<sub>1</sub> score is a metric that evaluates the accuracy of a model on a given dataset. It combines precision and recall measures and is calculated as the harmonic mean of these two values [13]. The F<sub>1</sub> score is particularly useful when comparing and assessing different models that make predictions about the same thing.

**Precision**—Precision is a measure that represents the proportion of relevant results [14]. It quantifies the ratio of accurately classified positive examples to the total number of positively classified examples.

**Recall**—Recall, also known as sensitivity or true positive rate, represents the percentage of correctly classified relevant results by the algorithm [15]. It quantifies the ratio of accurately classified positive examples to the total number of actual positive examples.

**ROC-AUC**—The ROC curve plots the relationship between the true positive rate (rate of actual positive examples) and the false positive rate (rate of false positive examples). The AUC provides a numerical value that indicates the overall

performance of the classifier in distinguishing between positive and negative examples.

The performance evaluation of each prediction model includes metrics such as accuracy, precision, recall,  $F_1$  score, and the ROC-AUC curve. The ROC-AUC score serves as a key criterion for selecting the final model. The results of the performance evaluation are summarized in the subsequent subsections.

## 4 Results

In this section, a comprehensive analysis of the observations from various regression methods is presented. The dataset [7] was collected for the experiments, and the features used in the analysis are listed in Table 1.

### 4.1 Decision Tree

The following Table 2 presents the outcomes of the decision tree analysis, considering various hyperparameters. These include the splitting criterion (entropy and Gini index) and the number of folds (3, 5, and 10) for the stratified K-fold cross-validation. Based on the Table 2, the decision tree achieves the highest  $F_1$  score of 0.94218 when employing the entropy criterion for splitting and tenfold cross-validation. Additionally, the highest ROC-AUC of 0.89232 is observed when using the Gini index as the splitting criterion and five folds as the hyperparameters for training the model.

**Table 1** Features used in the experiments from the dataset [10]

Previous	Episiotomy	Start week of antenatal care	Weight
Complications	Oxytocin	ART	BMI
Robson group	Obstetric risk	Previous term pregnancies	Age
ART mode	Parity	Amniotic liquid	Cardiotocography
Previous preterm pregnancies	Fatal intrapartum pH	Number of miscarriages	Maternal education
Amniocentesis	Comorbidity	Anaesthesia	Substance abuse
Pre-induction	Number of previous	Gestational stage	Smoking
Induction	Weight increased during pregnancy	Height	Alcohol

**Table 2** Observations of the decision tree method

Decision tree		Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
Criterion	K-folds					
Entropy	3	0.81007	0.82340	0.92919	0.93308	0.88431
	5	0.79870	0.83902	0.93991	0.93861	0.89066
	10	0.80915	0.80878	0.93276	0.94218	0.88977
Gini index	3	0.78305	0.80389	0.92919	0.93227	0.88783
	5	<b>0.79651</b>	<b>0.81268</b>	<b>0.93422</b>	<b>0.93276</b>	<b>0.89232</b>
	10	0.79586	0.80677	0.93421	0.93877	0.88898

## 4.2 Random Forest

According to the results given in Table 3, the random forest model achieves the highest  $F_1$  score of 0.95355 when using the entropy criterion, 50 estimators, and tenfold cross-validation. Additionally, the highest ROC-AUC value of 0.98513 is observed when utilizing the entropy criterion, 50 estimators, and five folds for training the model.

## 4.3 K-Nearest Neighbour

The following Table 4 gives the results of KNN considering different hyperparameters such as P (1 for Manhattan distance measure and 2 for Euclidean measure), number of neighbours (10, 15, 20, 25) and number of folds (3, 5, 10) for layered K-fold cross-validation folds. Based on the results presented in Table 4, the KNN model achieves the highest  $F_1$  score of 0.91278 when using Manhattan as the distance measure, 10 neighbours, and tenfold cross-validation. Moreover, the highest ROC-AUC value of 0.96037 is observed when utilizing Manhattan as the distance measure, 25 neighbours, and five folds for training the model.

## 4.4 Support Vector Machine

Table 5 gives the SVM results considering different hyperparameters such as kernel (linear, RBF, polynomial) and many convolutions (3, 5, 10) for the stratified K-fold cross-validation. From the results given in Table 5, the SVM model achieves the highest  $F_1$  score of 0.94721 when using the RBF kernel with tenfold cross-validation. Additionally, the highest ROC-AUC value of 0.97358 is observed when using the RBF kernel and ten convolutions as hyperparameters for training the model [15].

**Table 3** Observations of the random forest classification method

Random forest			Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
Criterion	Number of estimators	K-folds					
Entropy	20	3	0.85336	0.85366	0.95176	0.95030	0.97976
		5	0.85811	0.86146	0.95160	0.95257	0.98154
		10	0.86301	0.84975	0.95030	0.95046	0.98141
	30	3	0.85407	0.85366	0.95160	0.95274	0.98119
		5	0.85544	0.86146	0.95322	0.95111	0.98251
		10	0.85271	0.85856	0.95485	0.95225	0.98258
	40	3	0.85711	0.84487	0.95095	0.95290	0.98354
		5	0.86522	0.86049	0.95128	0.95241	0.98400
		10	0.85792	0.86341	0.95338	0.95338	0.98451
	50	3	0.85595	0.85952	0.95241	0.95257	0.98259
		5	0.85387	0.86341	0.95241	0.95274	0.98261
		10	<b>0.85808</b>	<b>0.86542</b>	<b>0.95306</b>	<b>0.95355</b>	<b>0.98513</b>
Gini index	20	3	0.85217	0.84682	0.95030	0.94786	0.98035
		5	0.86662	0.85171	0.95160	0.95209	0.97829
		10	0.85747	0.85371	0.95111	0.95030	0.98112
	30	3	0.85535	0.85074	0.95225	0.94998	0.98152
		5	0.85570	0.85854	0.95274	0.95290	0.98058
		10	0.85671	0.86726	0.95160	0.95241	0.98173
	40	3	0.85148	0.85756	0.95209	0.95127	0.98248
		5	0.85651	0.85659	0.95339	0.95209	0.98249
		10	0.85343	0.87223	0.95225	0.95062	0.98395
	50	3	0.86032	0.85269	0.95192	0.95063	0.98262
		5	0.85409	0.86829	0.95144	0.95306	0.98355
		10	0.85369	0.86828	0.95225	0.95160	0.98340

#### 4.5 AdaBoost

Table 6 gives the results of AdaBoost considering various hyperparameters such as decision tree and random forest as base estimators, learning rate (0.01, 0.1, 1), number of estimators (20, 30, 40, 50) and number of folds (3, 5, 10) for stratified K-fold cross-validation folds. The highest F1 score for AdaBoost is 0.96768 when the random forest is considered the base estimator, with a learning rate of 0.1, 50 estimators, and tenfold cross-validation. The highest ROC-AUC is 0.98768 when the random forest is used as the base estimator, along with a learning rate of 0.1, 50 estimators, and 10 folds as hyperparameters for training the model [16].

**Table 4** Observations of the KNN classification method

K-nearest neighbour			Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
$P$ value (Minkowski)	Number of neighbours	K-folds					
$P = 1$ (Manhattan)	10	3	0.69348	0.84098	0.91506	0.91018	0.94515
		5	0.69764	0.82732	0.91327	0.90937	0.94840
		10	0.69943	0.83417	0.91343	<b>0.91278</b>	0.94992
	15	3	0.66243	0.87122	0.89963	0.90190	0.95339
		5	0.65373	0.86537	0.90418	0.90109	0.95452
		10	0.65067	0.87127	0.90271	0.90320	0.95515
	20	3	0.67103	0.86830	0.90206	0.90612	0.95716
		5	0.90921	0.87122	0.90921	0.90872	0.95902
		10	0.68313	0.87322	0.90758	0.91100	0.95799
	25	3	0.65111	0.88878	0.89654	0.90109	0.95995
		5	0.65088	0.88195	0.90255	0.90125	<b>0.96037</b>
		10	0.65184	0.88201	0.90320	0.90304	0.95947
$P = 2$ (Euclidean)	10	3	0.58807	0.86439	0.87299	0.87526	0.93629
		5	0.58136	0.86537	0.87786	0.87397	0.93750
		10	0.58990	0.87612	0.87608	0.87494	0.93982
	15	3	0.54066	0.90147	0.85578	0.85951	0.94375
		5	0.54588	0.89268	0.86081	0.85903	0.94228
		10	0.55094	0.89567	0.86146	0.85903	0.94307
	20	3	0.54879	0.90048	0.86260	0.86048	0.94354
		5	0.55058	0.89268	0.85773	0.85951	0.94547
		10	0.56096	0.89079	0.86422	0.86422	0.94498
	25	3	0.52609	0.90342	0.84717	0.84717	0.94654
		5	0.52920	0.90341	0.85058	0.84847	0.94673
		10	0.53865	0.90446	0.85366	0.85301	0.94786

Table 7 gives the performances of five classifiers using stratified K-fold cross-validation. The models were evaluated based on various performance parameters, including precision, recall score,  $F_1$  score, and ROC-AUC. The shortlisted results are determined by the ROC-AUC score across different models and parameters.

In summary, the final observation reveals that the machine learning technique AdaBoost achieves the highest ROC-AUC score of 0.98532. It utilizes the random forest base estimator along with a learning rate of 0.1, 30 estimators, and ten convolutions. The second-highest score of 0.98513 is obtained by the random forest technique. This indicates that machine learning techniques outperform other methods in the ensemble. It is worth noting that due to the utilization of the cross-validation technique, there may be slight variations in accuracy, precision, recall,  $F_1$  score, and ROC-AUC score in each run of the model shown in the Fig. 7.

**Table 5** Observations of the SVM method

Support vector machine (SVM)		Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
Kernel	K-folds					
Linear	3	0.76637	0.90730	0.93877	0.94234	0.97121
	5	0.77390	0.91220	0.97121	0.93714	0.97148
	10	0.77041	0.90726	0.93584	0.93682	0.97139
RBF	3	0.80859	0.89952	0.94803	0.94721	0.97190
	5	0.80474	0.90049	0.94673	0.94575	0.97235
	10	<b>0.80274</b>	<b>0.89464</b>	<b>0.94591</b>	<b>0.94543</b>	<b>0.97358</b>
Poly	3	0.75292	0.90829	0.93373	0.93666	0.97044
	5	0.76400	0.90829	0.93341	0.93341	0.97098
	10	0.74731	0.91021	0.97098	0.93324	0.97040

**Table 6** Observations of the AdaBoost method

AdaBoost				Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
Base estimator	Learning rate	Number of estimators	K-folds					
Decision tree	0.01	20	3	0.78566	0.80777	0.93325	0.93357	0.88587
			5	0.78558	0.81366	0.93649	0.93308	0.88813
			10	0.80129	0.84773	0.93438	0.93487	0.89486
		30	3	0.79744	0.82046	0.93243	0.93422	0.88070
			5	0.78889	0.82439	0.93097	0.93520	0.88247
			10	0.79050	0.82054	0.93519	0.93227	0.88021
	40	30	3	0.77859	0.80875	0.93178	0.93162	0.88442
			5	0.78996	0.81659	0.93649	0.93503	0.88422
			10	0.80587	0.82145	0.93178	0.93422	0.88168
	50	30	3	0.78195	0.82339	0.93097	0.92415	0.89212
			5	0.79240	0.83610	0.93649	0.93065	0.88169
			10	0.79191	0.82147	0.93162	0.93649	0.88507
	0.1	20	3	0.78681	0.80095	0.93438	0.93114	0.88928
			5	0.79156	0.80878	0.93422	0.93406	0.88530
			10	0.79015	0.82046	0.93194	0.93324	0.88665
		30	3	0.77273	0.81949	0.93146	0.92837	0.88608
			5	0.78911	0.80488	0.93243	0.93211	0.89018

(continued)

**Table 6** (continued)

AdaBoost				Precision	Recall	Accuracy	<i>F</i> <sub>1</sub> score	ROC-AUC	
Base estimator	Learning rate	Number of estimators	K-folds						
Random forest	0.01	40	10	0.80627	0.80971	0.93178	0.93210	0.88401	
			3	0.78448	0.81265	0.93097	0.93357	0.88382	
			5	0.78775	0.80683	0.93633	0.93438	0.88500	
			10	0.79650	0.81652	0.93714	0.93422	0.89329	
		50	3	0.77861	0.82047	0.93243	0.93260	0.88490	
			5	0.78648	0.81073	0.93130	0.93243	0.88940	
			10	0.79306	0.82824	0.93081	0.93487	0.89190	
		1	20	3	0.78555	0.80584	0.93341	0.93146	0.88704
			5	0.79326	0.81854	0.93536	0.93227	0.88393	
			10	0.79799	0.83024	0.93259	0.93568	0.88448	
			30	3	0.77964	0.81557	0.93130	0.93292	0.88675
			5	0.79223	0.81561	0.93373	0.93308	0.88432	
			10	0.79169	0.80191	0.93600	0.93145	0.88851	
		50	20	3	0.76881	0.81562	0.92821	0.93406	0.88509
			5	0.79311	0.81659	0.93422	0.93373	0.88091	
			10	0.80847	0.81849	0.93584	0.93324	0.88791	
			30	3	0.77670	0.82536	0.93000	0.93016	0.88208
			5	0.79057	0.83024	0.92886	0.93406	0.88120	
			10	0.80540	0.81261	0.93584	0.93357	0.89100	
		0.1	20	3	0.84823	0.86829	0.95241	0.95176	0.98336
			5	0.84801	0.86732	0.95176	0.95225	0.98390	
			10	0.85256	0.86734	0.95111	0.95192	0.98470	
			30	3	0.85258	0.86439	0.95306	0.95192	0.98355
			5	0.85217	0.87024	0.95371	0.95274	0.98379	
			10	0.85085	0.86735	0.95127	0.95322	0.98513	
			40	3	0.84699	0.86244	0.95290	0.95241	0.98406
			5	0.84913	0.86341	0.95241	0.95176	0.98427	
			10	0.84955	0.86637	0.95290	0.95241	0.98356	
			50	3	0.85121	0.86244	0.95274	0.95225	0.98344
			5	0.84985	0.86634	0.95274	0.95306	0.98398	
			10	0.84976	0.86446	0.95176	0.95144	0.98457	
		0.1	20	3	0.85103	0.86340	0.95225	0.95128	0.98326
			5	0.84481	0.86732	0.95306	0.95290	0.98418	
			10	0.85112	0.86735	0.95257	0.95257	0.98470	
			30	3	0.85380	0.86049	0.95160	0.95111	0.98403
			5	0.84878	0.86439	0.95111	0.95176	0.98483	

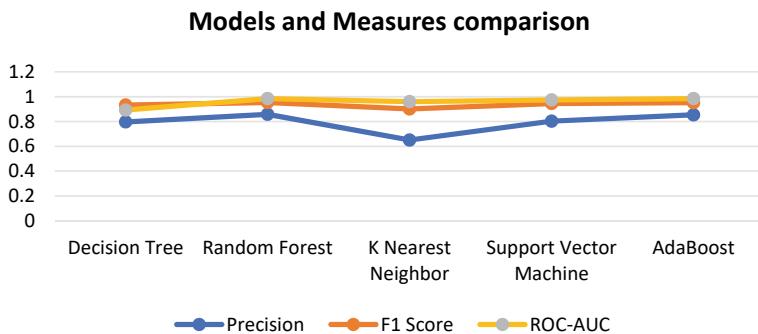
(continued)

**Table 6** (continued)

AdaBoost				Precision	Recall	Accuracy	$F_1$ score	ROC-AUC	
Base estimator	Learning rate	Number of estimators	K-folds						
		40	10	<b>0.85418</b>	<b>0.86636</b>	<b>0.95225</b>	<b>0.95144</b>	<b>0.98532</b>	
			3	0.85164	0.86342	0.95274	0.95176	0.98410	
			5	0.85135	0.87317	0.95274	0.95176	0.98367	
		50	10	0.85168	0.86732	0.95355	0.95144	0.98509	
			3	0.85457	0.86927	0.95225	0.95192	0.98372	
			5	0.85200	0.86829	0.95225	0.95144	0.98440	
		1	10	0.85099	0.86149	0.95306	0.95273	0.98449	
			20	3	0.85209	0.86535	0.95290	0.95063	0.98317
			5	0.85134	0.86049	0.95176	0.95306	0.98429	
		30	10	0.85817	0.86637	0.95192	0.95176	0.98407	
			3	0.85736	0.86244	0.95241	0.95322	0.98425	
			5	0.85018	0.85951	0.95111	0.95322	0.98470	
		40	10	0.84864	0.86440	0.95273	0.95225	0.98481	
			3	0.85253	0.86732	0.95387	0.95290	0.98394	
			5	0.84557	0.86049	0.95404	0.95192	0.98426	
		50	10	0.85124	0.86244	0.95338	0.95192	0.98508	
			3	0.84705	0.86145	0.95241	0.95079	0.98451	
			5	0.85227	0.86146	0.95209	0.95322	0.98407	
			10	0.85321	0.87125	0.95290	0.95192	0.98517	

**Table 7** Comparative results of the regression methods

Model used	Precision	Recall	Accuracy	$F_1$ score	ROC-AUC
Decision tree	0.79651	0.81268	0.93422	0.93276	0.89232
Random forest	0.85808	0.86542	0.95306	0.95355	0.98513
K-nearest neighbour	0.65088	0.88195	0.90255	0.90125	0.96037
Support vector machine	0.80274	0.89464	0.94591	0.94543	0.97358
AdaBoost	0.85418	0.86636	0.95225	0.95144	0.98532



**Fig. 7** Graph comparing  $F_1$  scores and ROC-AUC score of all the models

## 5 Conclusion

In conclusion, the selection of appropriate delivery methods is crucial for ensuring the safety of both the mother and the new-born child. This study aimed to explore the optimal characteristics for predicting the mode of delivery by investigating various features and employing different models. We compared traditional classifiers, including decision tree, KNN, and SVM, with ensemble classifiers such as AdaBoost and random forest, which are innovative techniques for birth mode prediction. Through this analysis, clinicians and users were able to identify significant factors that influence the likelihood of a cesarean section.

By evaluating the performance of all algorithms using different performance parameters such as precision, recall,  $F_1$  score, and ROC-AUC, we shortlisted the models based on the ROC-AUC score across various parameter combinations. Our findings revealed that the AdaBoost model achieved the highest ROC-AUC score of 0.98532 when trained with a random forest base estimator, a learning rate of 0.1, 30 estimators, and ten convolutions. The second-highest ROC-AUC score of 0.98513 was obtained by the random forest model.

Overall, the results indicate that ensemble machine learning models, such as AdaBoost and random forest outperform traditional machine learning models in predicting the mode of delivery. These findings contribute to the ongoing efforts in enhancing the decision-making process for clinicians and improving the safety and well-being of both mothers and new-borns.

## References

- Alam SMB, Patwary MJA, Hassan M (2021) Birth mode prediction using bagging ensemble classifier: a case study of Bangladesh. In: 2021 International conference on information and communication technology for sustainable development (ICICT4SD)

2. Harrison MS, Garces AL, Goudar SS et al (2020) Cesarean birth in the global network for women's and children's health research: trends in utilization, risk factors, and subgroups with high cesarean birth rates. *Reprod Health* 17(Suppl 3):165
3. Rahman S et al (2021) Risk prediction with machine learning in cesarean section: optimizing healthcare operational decisions. *Sig Process Tech Comput Health Inf* 293–314
4. Islam MN, Mahmud T, Khan NI, Mustafina SN, Islam AKMN (2016) Exploring machine learning algorithms to find the best features for predicting modes of Childbirth. In: 2016 International conference on computing communication control and automation (ICCUBEA)
5. Abbas S, Riaz R, Kazmi S, Rizvi S, Kwon S (2018) cause analysis of cesarean sections and application of machine learning methods for classification of birth data, pp 1–1. IEEE Access. <https://doi.org/10.1109/ACCESS.2018.2879115>
6. Islam MN, Mahmud T, Khan NI, Mustafina SN, Najmul Islam AKM (2021) exploring machine learning algorithms to find the best features for predicting modes of Childbirth. IEEE Access
7. Campillo-Artero C, Serra-Burriel M, Calvo-Pérez A (2018) Predictive modeling of emergency cesarean delivery. *PLoS ONE* 13(1):e0191248
8. Wie JH, Lee SJ, Choi SK, Jo YS, Hwang HS, Park MH, Kim YH, Shin JE, Kil KC, Kim SM, Choi BS, Hong H, Seol H-J, Won H-S, Ko HS, Na S (2022) Prediction of emergency cesarean section using machine learning methods: development and external validation of a nationwide multicenter dataset in the Republic of Korea. *Life* 12:604
9. Jamjoom MM (2020) Data mining in healthcare to predict cesarean delivery operations using a real dataset. In: First international conference on computing and emerging sciences ICCE, vol 2020
10. Jijo B, Mohsin Abdulazeez A (2021) Classification based on decision tree algorithm for machine learning. *J Appl Sci Technol Trends* 2:20–28
11. Chen RC et al (2020) Selecting critical features for data classification based on machine learning methods. *J Big Data* 7(1):52
12. Taunk K et al (2019) A brief review of nearest neighbor algorithm for learning and classification. In: 2019 International conference on intelligent computing and control systems (ICCS). IEEE
13. Powers DMW (2020) Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. [arXiv:2010.16061](https://arxiv.org/abs/2010.16061) [cs.LG]
14. Cano Lengua MA, Papa Quiroz EA (2020) A systematic literature review on support vector machines applied to classification. In: 2020 IEEE engineering international research conference (EIRCON), Lima, Peru, 2020, pp 1–4.<https://doi.org/10.1109/EIRCON51178.2020.9254028>
15. Thammasiri D, Meesad P (2012) Adaboost ensemble data classification based on diversity of classifiers. *Adv Mater Res* 403–408:3682–3687
16. Hatwell J, Gaber, MM, Atif Azad RM, Ada-WHIPS: explaining AdaBoost classification with applications in the health sciences. *BMC Medical Informatics and Decision Making*, 20, 250 (2020). <https://doi.org/10.1186/s12911-020-01201-2>

# Novel Hybrid Methods for Journal Article Summarization Combining Graph Method and Rough Set TFIDF Method with Pegasus Model



K. Sheena Kurian and Sheena Mathew

**Abstract** Journal article summarization shortens an article, including all the relevant and significant topics. As the number of journal articles published every year is increasing rapidly, the significance of research on journal article summarization also increases. This work generates a summary of journal articles from the ScisummNet dataset using extractive, abstractive and novel hybrid methods. The summary generated by these methods is then analyzed using word overlap and semantic similarity scores to find the best summary. The precision, recall and f-measure scores of rouge-1, rouge-2, rouge-L and rouge-Lsum scores are analyzed to find the word overlap score and precision, recall and f-measure of the BERT score to find the semantic similarity score. The summary generated by the novel hybrid methods combining the extractive methods like the graph with the sum of weights, the graph with bushy path and the rough set TFIDF methods with the abstractive Pegasus fine-tuned model gives the best rouge and BERT scores among the other hybrid methods.

**Keywords** Abstractive · Fine-tuning · Graph methods · Rough set · Pre-trained models · Summarization

## 1 Introduction

Journal article summarization is an extensive area of research in natural language processing (NLP) to generate a concise, coherent and readable summary of an article conveying all the salient topics. A lot of published articles are available now, and it is difficult for readers to choose the best one for detailed study and understanding. A good summary of salient sentences can speed up their research investigation as they can easily decide on the article for elaborate reading. Though the journal articles have an author-written abstract along with the paper, it may not be the best possible

---

K. Sheena Kurian (✉) · S. Mathew

School of Engineering, Cochin University of Science and Technology, Kerala, India  
e-mail: [sheenakurian@gmail.com](mailto:sheenakurian@gmail.com)

S. Mathew  
e-mail: [sheenamathew@cusat.ac.in](mailto:sheenamathew@cusat.ac.in)

summary that includes all the salient information. Journal articles are usually long, with 3000 or more tokens. The compression ratio is high for long document summarization, and information loss is more. There is no layout bias in the case of a long document. Journal articles have different sections, each with a specific purpose. A good summary of journal articles should contain salient information from all of these sections.

Text summarization can be done using extractive, abstractive and hybrid methods [1]. Extractive methods rank the sentences in the article and choose the best-ranked sentences. The chosen sentences are ordered sequentially as per the original article. Though each selected sentence is grammatically correct, the summary may lack fluency, coherence and readability. Abstractive methods comprehend the concepts of the article and rephrase them. However, it results in a more fluent, concise, non-redundant, semantically coherent and readable summary with good linguistic quality. Abstractive methods are more complicated and have been explored less than extractive methods until recently. It requires domain-specific ontology for analysis and the salience computation of concepts. The design and development of pre-trained abstractive models contributed to the research in abstractive and hybrid summarization methods. But they have a limitation that they can process only small documents. Most pre-trained state-of-the-art abstractive models are limited to 512 or 1024 tokens. The abstractive model architecture requires a novel design to overcome the hardware limitations and to be useful for journal article summarization. In the hybrid methods considered in this work, the summary generated by extractive methods is input to abstractive models. This paper analyzes the performance of extractive, abstractive and hybrid methods in generating the summary of journal articles selected from the ScisummNet dataset [2] using rouge [3] and BERT score [4]. The rouge scores find the word overlap, and the BERT scores find the semantic similarity between the summary generated and the gold summary available in the dataset. These scores can identify whether the summary generated is factually consistent with the source document. The best rouge scores for the extractive methods were for the graph with the sum of weights method, graph with bushy path method and rough set TFIDF methods [5]. The abstractive summary is generated using BART, T5, Pegasus pre-trained, Pegasus fine-tuned, BigBirdPegasus and GPT3 models. The best scores were for the BART and Pegasus fine-tuned models. In the hybrid methods, the summary generated by the graph with the sum of weights, the graph with bushy path and rough set TFIDF methods is input to the state-of-the-art abstractive models like BART and Pegasus fine-tuned models.

The remaining sections of the paper are as follows. Section 2 explains some salient research works and datasets used in text summarization. Section 3 explains the motivation for this work and the main challenges faced in the process. The extractive, abstractive and hybrid methods implemented in this work are detailed in Sect. 4. Section 5 gives the results of the experiments, and Sect. 6 presents the conclusion and future works.

## 2 Related Works and Summarization Datasets

This section discusses salient works on text summarization. There are extractive, abstractive and hybrid methods to solve the problem of text summarization. The graph-based extractive approach uses the mutual relationship of sentences to rank and select sentences for the summary. The graph is constructed with sentences, words or paragraphs as nodes and relationships between these nodes as edges [6–8]. The rough set concept of reducts and core can also be used for summarization [9]. Sequence-to-sequence (seq2seq), neural attention, convolutional attention-based encoder, conditional recurrent neural network decoder, graph-based attention in the encoder-decoder framework and heterogenous graph neural networks have gained popularity in neural abstractive text summarization [10–13]. Transformer models increase the efficiency of handling and comprehending sequential data for text summarization. It can process the input data parallelly, whereas the previous RNN models were sequential. Bidirectional Encoder Representations from Transformers (BERT) is one of the advanced Transformer-based models pre-trained on over 2500 million words from Wikipedia and 800 million words from BookCorpus [14]. The maximum input size is around 512 tokens. So a long input is broken into smaller segments before applying them as input. This content fragmentation also causes a significant loss of context. BART [15], Pegasus [16] and T5 [17] models are also used for generating an abstractive summary (explained in Sect. 4). But Transformer-based abstractive models often generate repetitive, ungrammatical and factually inconsistent summaries [18, 19]. Longformer replaces Transformer’s self-attention mechanism with dilated sliding window attention to reduce computation and memory usage [20]. BigBird runs on a sparse attention mechanism that allows it to overcome the quadratic dependency of BERT while preserving the properties of full-attention models. In addition to sparse attention, BigBird applies global and random attention to the input sequence [21]. BigBird uses a sparse attention mechanism which enables it to process sequences of length up to 4096 tokens using the same hardware as BERT. HAT adds hierarchical attention layers to an encoder-decoder model to summarize long documents [22]. Extractive and abstractive techniques are combined to create a better summary [23, 24].

The DUC dataset consisting of news articles paired with human-written summaries is the de facto standard the NLP community uses for evaluating summarization methods. SCIRLDR dataset has TLDRs (main points summarizing the longer explanation) written by human experts and authors of computer science articles from OpenReview [25]. The Wikipedia-summary dataset consists of text extracted from Wikipedia [26]. SumPubMed dataset has scientific articles from PubMed archive with human analysis of summary coverage, redundancy, readability, coherence and informativeness [27]. The arXiv dataset from the arXiv open access repository has the abstract as the reference summary [28]. ScisummNet is the first large-scale human-annotated dataset of 1040 journal articles in computational linguistics and the NLP domain, developed as part of the CL-SciSumm Shared Task. The dataset has 60 anno-

tated sets of citing and reference papers and the three types of summaries (abstract, community summary collated from reference spans of its citances and summary by annotators) [2].

### 3 Motivation and Challenges in Article Summarization

Journal articles provide in-depth research findings and discussions. A summary provides a quick overview of the salient points. It helps readers to screen and select relevant articles for detailed reading. Summaries make scholarly articles reach a broader audience and facilitate quick collaboration between researchers and practitioners in various domains. The main challenges in this area are:

- The summary generated may contain less salient information to a particular reader. Different readers may be reading an article from different perspectives. No single summary can meet the requirements of all readers.
- Salient content in journal articles is distributed throughout the article, whereas it is toward the beginning in news articles. Journal articles contain rare domain-specific scientific terms. So the summarization methods that perform well for news articles may not be that promising for journal articles.
- The summary generated may contain less salient information to a particular reader. Different readers may be reading an article from different perspectives. No single summary can meet the requirements of all readers.
- Salient content in journal articles is distributed throughout the article, whereas it is toward the beginning in news articles. Journal articles contain rare domain-specific scientific terms. So the summarization methods that perform well for news articles may not be that promising for journal articles.
- Supervised long document datasets are unavailable. Few datasets are available with the abstracts as the gold summary. But an abstract does not include relevant content from the full article. It describes the viewpoint of the author. The human-written summary of a document is very different or more elaborate than the abstract written by the author.
- Lack of proper evaluation metrics to assess the correctness of the generated summary remains a challenge. Rouge is the most commonly used evaluation metric and is good at measuring the relevance of a sentence with the reference. The rouge score is low when the generated summary has new words not in the gold summary.
- Conversion of an article from pdf to text may have many errors, thereby questioning the correctness of the input we feed into the methods.
- Since extractive summarization methods select a sentence as a whole from the article, each sentence is grammatically correct. But, the summary formed from selected sentences may lack fluency and coherence.
- Multi-sentence summaries sometimes fail to make sense when referring to an entity by pronoun without first introducing it.

- The meaning of the newly generated phrases in the abstractive summary may not be the same as intended in the original document.
- Abstractive methods have been tested more on small documents. Scaling these methods to long documents is still a challenge. Scientific articles are considered long, with lots of domain-dependent terms, equations and figures.
- The summary generated by pre-trained abstractive models is sometimes factually inconsistent with the article.

## 4 Methodology

Extractive, abstractive and hybrid methods for journal article summarization are implemented in this work. In extractive summarization, the salient sentences from the article are selected. The proposed rough set TFIDF and graph methods use the extractive strategy. In the abstractive method, the summary is framed by rephrasing the concepts using some pre-trained abstractive models with suitable parameters for the dataset and then fine-tuned to obtain better results. In the hybrid method, novel methods are proposed and implemented by inputting the summary generated by the extractive methods using graph and rough set TFIDF methods into the abstractive models to get better results. The word overlap and semantic similarity scores of the different methods are evaluated in this work.

### 4.1 Extractive Summarization Methods

Extractive methods use different ways to rank the sentences in the document. The highly ranked sentences are selected and arranged in the order of its appearance in the original document to get the summary. The graph with sum of weights, graph with bushy path and rough set TFIDF methods have the highest rouge scores [5], and hence, they are selected for further study in this work.

**Graph Methods** The graph methods create a graph consisting of vertices and edges. The vertices may represent paragraphs, sentences or words in the article, and the edges are the relationship between the vertices. The edge weights could be the count or TFIDF scores of the common words between the sentences (vertices). Two variants of the graph methods are proposed and implemented in this work for ranking the vertices: the graph with the sum of edge weights and the graph with bushy path methods. In both methods, the top-ranking vertices are selected to form the summary. The ranking algorithms are explained below.

1. Graph with Sum of Edge Weights Method: The proposed algorithm uses the sum of edge weights of graph vertices as the selection criterion. Here, a vertex is scored based on the sum of common words a vertex shares with other vertices. The different steps in the algorithm are:

- (a) Read the text. Remove the stop words. Remove sentences with special characters or equations and sentences with less than three words.
  - (b) Create sentence to sentence matrix with each matrix entry as the number of common words in both sentences.
  - (c) Construct a graph with sentences as vertices and the word counts as the edge weights of the graph.
  - (d) Find the row sum (sum of edge weights of all edges connected to a vertex).
  - (e) Sort the row sum. Select a required number of vertices (sentences) with the top row sum values into a list.
  - (f) The selected sentences are arranged in the order in which they appear in the original text.
2. Graph with Bushy Path Method: A vertex is scored based on the number of edges having common words with other vertices above a threshold value. The proposed algorithm for generating a summary with the bushy path of a graph as the selection criterion:
- (a) Read the text. Remove the stop words. Remove sentences with special characters or equations and sentences with less than three words.
  - (b) Create sentence to sentence matrix with each matrix entry as the number of common words in both sentences.
  - (c) Construct a graph with sentences as vertices of the graph. The word counts are the edge weights of the graph.
  - (d) If an edge weight is greater than the threshold  $w$ , increment the score of both vertices by one. Repeat this process with all edges in the graph.
  - (e) Sort the vertex scores and select vertices of top scores to the summary list till the required length. The vertex with a high score is connected to many other vertices. The interconnections indicate a bushy path. It means that the sentence discusses an important topic.
  - (f) The vertex numbers in the summary list are sorted in increasing order to arrange the sentences in the order in which they appear in the original text.

The four proposed implementations of graph methods here are the graph with the sum of weights and the graph with bushy path methods generating a summary of 30 and 70% length of the article. The 30% length summary has high precision and f-measure compared to the 70% length. The 70% summary has high recall values of rouge scores. The graph with the sum of weights method performs slightly better than the graph with the bushy path method (more by an average of 0.083 for 30% and 0.003 for 70% in f-measure). Details are given in Sect. 5.

**Rough Set TFIDF Methods** The rough set TFIDF method is an extractive summarization technique, and the proposed algorithm is as follows:

- (a) Split the journal article into sentences. Remove stop words and tokenize the sentence into words.
- (b) The word-sentence matrix (information table) is created with the unique words in the article. The table entry is one if the word is present in the sentence and

zero otherwise. Words represent the rows (objects), and sentences represent the columns (attributes).

- (c) The size of the information table increases with the article length. Removing words occurring in only one sentence reduces the size [9].
- (d) By partitioning the words based on their co-occurrence in sentences, we identify equivalence classes of words in the information table. Some equivalence classes have all sentences in the summary group (positive class) while others in the non-summary group (negative class). The equivalence relation may not distinguish summary and non-summary sentences if two sentences with similar words have one sentence in the summary class and the other in the non-summary class. The rough sets can handle this incomplete information.
- (e) The reducts are the minimal set of attributes preserving the positive class and are created from the equivalence classes by omitting one sentence at a time.
- (f) The core is found by finding the intersection of sentences in the reducts. When the core is small or empty, the attributes repeated in the maximum number of reducts are takencite13Sheena2023.

Three variants are proposed and implemented here: the rough set core, the rough set fuzzy core and the rough set TFIDF method. The rough set TFIDF method had the highest rouge score and is taken for further analysis [5]. In the rough set TFIDF method, word-frequency counts are found from the word-sentence matrix by adding the row entries and are then normalized. Sentences are given scores by adding the normalized frequency count of all words in the sentence. The summary of the article is formed by adding top-scored sentences for the required length [5]. The normalized frequency count of a word and the sentence score is computed using Eqs. 1 and 2. Rough set TFIDF methods are implemented to generate a summary of 30% and 70% of the length of the article. The 30% summary has high precision and f-measure compared to the 70% summary. The 70% summary has high recall values. Details are shown in Sect. 5.

$$\text{Normalized freq. count of a word} = \frac{\text{No. of sentences with the word}}{\text{Total no. of sentences}} \quad (1)$$

$$\text{Sentence score} = \sum (\text{Normalized freq. count of words in the sentence}) \quad (2)$$

## 4.2 Abstractive Summarization Methods

Abstractive text summarization is one of the most challenging tasks in NLP. Abstractive methods understand the article and generate paraphrased summary with the main points. Transformer models are the state-of-the-art in seq2seq problems like text summarization [14]. They are categorized into encoder, decoder and encoder-decoder models. BERT, Roberta and Albert are popular encoder models. GPT2 and GPT3 are the popular decoder models, and T5, BART and BigBird are encoder-decoder models. The Hugging Face library implementation for Transformer-based

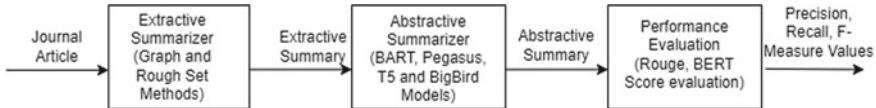
encoder-decoder models like BART, T5, Pegasus and BigBirdPegasus, and the OpenAI implementation of GPT3 is used here.

**BART Model** Bidirectional and Auto-regressive Transformer (BART) is a seq2seq model trained as a denoising autoencoder. The BART model by Facebook AI research combines Google’s BERT and OpenAI’s GPT. It is bidirectional like BERT and auto-regressive like GPT. BERT is a deeply bidirectional and unsupervised language representation pre-trained on a plain text corpus. The training data has noising schemes like token masking, token deletion, text infilling, sentence permutation and document rotation. The unsupervised pre-training of BART results in a language model. BART has nearly 140 million trained parameters and is fine-tuned on small labeled datasets to perform specific applications like text summarization [15]. The pre-trained facebook/bart-large-cnn model implementation of BART in the Hugging Face library is used in this work [29]. Details are shown in Sect. 5.

**T5-Large Model** Google’s T5 (Text-to-Text Transfer Transformer) is based on Transformer architecture using a text-to-text approach, where the input and output are both texts. It is used for supervised and unsupervised NLP tasks like machine translation, question answering, classification and summarization. T5 uses both encoder blocks and the decoder blocks of Transformers and is pre-trained on a 7TB Colossal Clean Crawled Corpus (C4) dataset. It is trained using teacher forcing. The T5 base model has 220 million trained parameters [17]. Hugging Face implementation of the pre-trained T5-large model is used in this work [29]. The no\_repeat\_ngram\_size parameter is assigned value 3, and length\_penalty to 2.0. Detailed results are shown in Sect. 5.

**Pegasus Models** Google’s Pegasus (Pre-training with Extracted Gap Sentences for Abstractive Summarization) is a pre-trained model that improves the performance of abstractive summarization methods. In Pegasus, salient sentences are removed/masked from the input document, and the model re-generates them from the remaining sentences. It is a difficult task for the model to solve perfectly. But, in the process, the model learns about the language and general facts about the world. This task closely resembles the fine-tuning process. The model generated sentences similar to the rest of the document to replace the masked sentences [16]. Two variants of the Pegasus model are implemented here.

1. **Pegasus Pre-trained Model:** The Hugging Face implementation of the pre-trained ‘google/pegasus-arxiv’ model is used in this work [29].
2. **Pegasus Fine-Tuned Model:** The Pegasus model required only a small number of examples for fine-tuning to get near state-of-the-art performance. The Hugging Face implementation of the Pegasus model is fine-tuned on the summarization dataset of 1000+ articles to improve the performance. The articles were hand-cleaned before fine-tuning as there was a lot of noisy text and symbols in the input. The fine-tuning program was executed for ten epochs with batch size = 1. The number of warmup steps for the learning rate scheduler was assigned to 50, and the weight-decay to 0.01.



**Fig. 1** Architecture of the hybrid model and performance evaluation

It is observed that the Pegasus fine-tuned model has a performance gain of 0.4 in the average of f-measure for rouge scores compared to the pre-trained model. Details are in Sect. 5.

**BigBirdPegasus** The BERT model works on self-attention mechanism on the entire input leading to quadratic growth of computational and memory requirements for every new input token. The BERT model has maximum input size of 512 tokens and is not ideal for scientific article summarization which has longer inputs [14]. Google's BigBird runs on a sparse attention mechanism token by token and overcomes the quadratic dependency of BERT. BigBird can process sequences of length 8x more than that with BERT, using the same hardware as BERT [21]. BigBird can handle sequences of 4096 tokens at a much lower cost. The Hugging Face implementation of BigBirdPegasus is used in this work. Two different variants of BigBirdPegasus are implemented here: BigBirdPegasus pre-trained model and BigBirdPegasus with no\_repeat Model. The no\_repeat\_ngram\_size parameter is set to 3 here to avoid repetition of trigrams in subsequent sentences. It is seen that there is a performance gain in rouge f-measure score by 0.04 on average for the summary generated by BigBirdPegasus with no\_repeat compared to BigBirdPegasus pre-trained. The details are given in Sect. 5.

**GPT3 Model** GPT3 is the third-generation generative pre-trained Transformer model. It has 175 billion learning parameters [30]. The OpenAI implementation of the GPT3 text-davinci-003 model is used here. The chunked text is concatenated with the Tl;dr command and given as the prompt input. Here, the temperature parameter is set to 0.7, top-p to 0.9, frequency-penalty to 0 and presence-penalty to 1. Details of the results are in Sect. 5.

### 4.3 Hybrid Summarization Methods

Vanilla pre-trained Transformer models for abstractive summarization can process only maximum of 1024 tokens. So more than half the input of a long document needs to be truncated for the abstractive models to work. Research indicates that in lengthy documents, the salient information is evenly distributed across the entire document than toward the beginning. So usage of abstractive models for long documents is not that promising. Hybrid methods effectively address the memory complexity and hardware limitations encountered in vanilla pre-trained abstractive models. In the hybrid methods experimented in this work, the summary generated by an extractive

method is given as input to an abstractive model, thereby reducing the burden on the abstractive model. The summary generated by the graph and rough set methods is input to high-performing abstractive models like BART and Pegasus fine-tuned models. It is seen that there is a significant improvement in performance for the hybrid models. Details are shown in Sect. 5. Figure 1 shows the general architecture of the hybrid models implemented in this work. The hybrid models suggested and experimented to identify the best method for journal article summarization in this work are explained here.

**Graph Methods Combined with Abstractive Models** The extractive summary generated by graph methods is input into the different abstractive summarization models to generate a summary. Four different variants of this hybrid method are experimented in this work. Extractive summary of 30% length of the article generated by (a) graph with the sum of weights method and (b) graph with bushy path method is given as input to the BART and Pegasus fine-tuned abstractive summarization models.

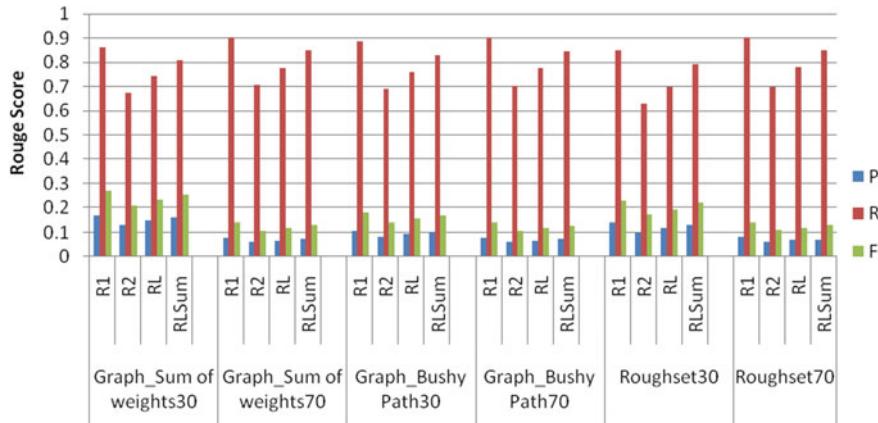
**Rough Set TFIDF Method Combined with Abstractive Models** The extractive summary generated by the rough set TFIDF method is input into the abstractive summarization models to generate a summary. Eight different variants of the hybrid method are experimented in this work. Extractive summary of 30 and 70% length of the article generated by rough set TDIDF method is input to abstractive models (a) BART, (b) Pegasus fine-tuned, (c) BigBirdPegasus and (d) BigBirdPegasus with no\_repeat\_ngram\_size parameter set to 3.

## 5 Results and Discussion

The scientific articles for the summary generation task in this work are from the ScisummNet dataset [2]. This data was hand-cleaned to remove noisy text so that the fine-tuning data is as clean as possible to get accurate results. Two types of intrinsic analysis assessing the quality of generated summary are carried out in this work [31]. The word overlap score is measured using rouge scores and the similarity score using the BERTscore.

### 5.1 Rouge Score for Word Overlap over Annotated Summary

Word overlap score is the number of common words in the generated and annotated summary. The precision, recall and f-measure of rouge-1 (R1), rouge-2 (R2), rouge-L (RL) and rouge-Lsum (RLsum) values are measured here to evaluate the generated summary [3]. Rouge-n measures the n-gram overlap, and rouge-L measures the longest common in-sequence of words between the generated and the gold summary. Rouge-Lsum is identical to rouge-L except that it considers newline also. The rouge

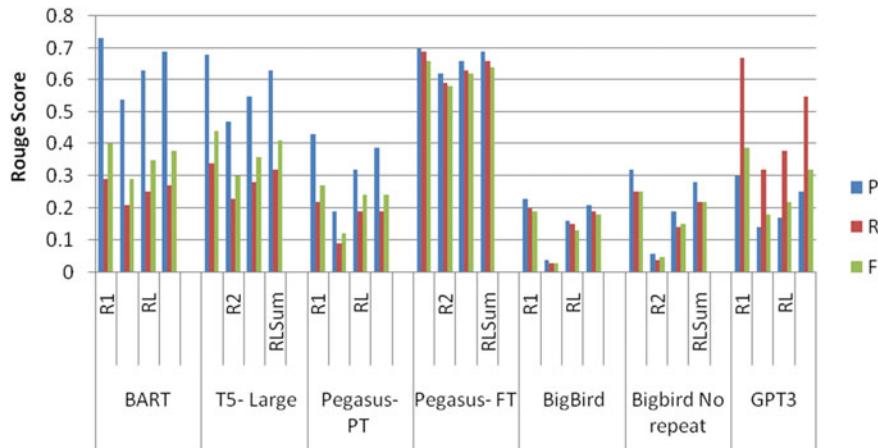


**Fig. 2** Rouge scores of extractive methods (graph and rough set TFIDF methods)

scores penalize the rephrasing or rewording of words in a summary that even a well-generated summary would only have lower scores. Also, rouge scores are not the best measures of readability, coherence or truth. Recall measures the completeness of the output compared to the gold summary. It indicates the number of words in the gold summary present in the generated summary. Recall=1 means that all the words in the reference summary are present in the generated summary. But other less salient words may also be present in the generated summary. Recall rewards for the number of words accurately predicted by the generated summary. Precision measures correctness based on the relevance of the retrieved information. It indicates how many words in generated summary are present in the gold summary and penalizes unnecessary verbosity of words in the generated summary. F-measure is a composite method combining precision and recall.

**Rouge Scores of Extractive Methods** Rouge scores of the generated summary by the extractive, abstractive and hybrid methods over human-annotated summary are analyzed. ScisummNet GitHub repository has 1000+ journal articles with human-annotated gold summaries. Figure 2 shows the rouge scores of the graph with the sum of weights, graph with bushy path and rough set TFIDF method generating 30% and 70% length extractive summary of the original article. The summary generated by the graph and rough set TFIDF methods has low precision values but high recall values.

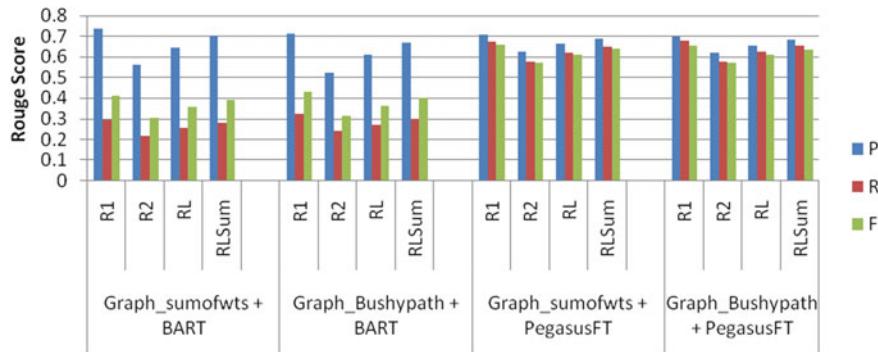
**Rouge Scores of Abstractive Methods** Figure 3 shows the rouge scores of the summary generated by BART, T5-large, Pegasus pre-trained, Pegasus fine-tuned, BigBirdPegasus, BigBirdPegasus with no-repeat and GPT3 abstractive models. BART generates high precision for the rouge scores compared to T5. There is a significant improvement in precision, recall and f-measure of rouge scores for the Pegasus fine-tuned model (70:30 ratio on training and validation set) over the Pegasus pre-trained



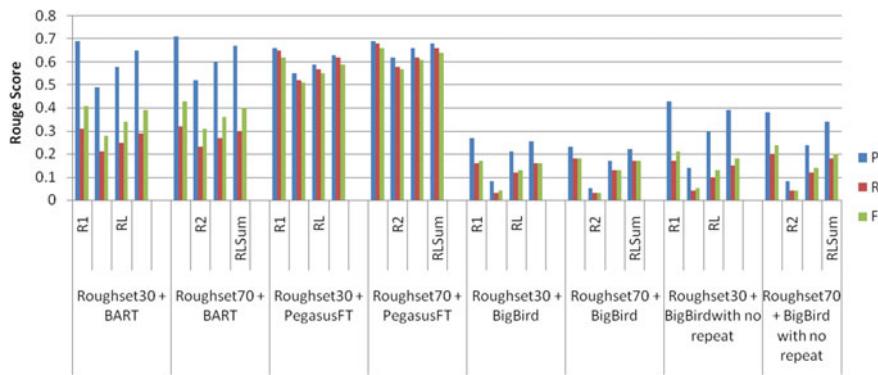
**Fig. 3** Rouge scores of abstractive models (BART, T5, Pegasus, BigBird, GPT3)

model. There is a notable improvement in rouge scores for BigBirdPegasus when no\_repeat\_ngram\_size = 3 (trigram repetition is avoided). GPT3 has relatively high recall values compared to precision.

**Rouge Scores of Hybrid Methods** Figure 4 shows the result of the hybrid summary generated by BART and Pegasus fine-tuned models when the extractive summary generated by the graph with the sum of weights and graph with bushy path method is the input. The abstractive summary generated by BART has a slightly better score when combined with the graph with the sum of weights method than with the graph with the bushy path method. The summary generated by the Pegasus fine-tuned method has similar scores when combined with the graph with the sum of weights or the graph with the bushy path method. Figure 5 shows the results when the extractive summary generated by rough set TFIDF method is input to the BART, Pegasus fine-tuned, BigBird and BigBird with no\_repeat summarizer models. BART takes only 1024 tokens as input. So there is only an increase of +0.01 to +0.02 in rouge scores when the summary of 70% length produced by the rough set TFIDF method is input to the BART summarizer than the summary of 30% length input to the BART. When the summary generated by rough set methods is input to the Pegasus models, there is an increase of +0.03 to +0.09 in rouge score values when a summary of 70% length generated by the rough set method is input to the fine-tuned Pegasus over 30% length summary input to fine-tuned Pegasus model. The output generated by BigBirdPegasus is short and less meaningful in many cases. The precision scores are improved when the summary of 70% length generated by the rough set TFIDF method is input to BigBirdPegasus with no\_repeat\_ngram\_size parameter set as 3. Results show that there is an improvement in P values for the hybrid model of the graph with sum of weights method and rough set TFIDF method generating 70% summary when combined with the Pegasus fine-tuned model compared to the extractive or



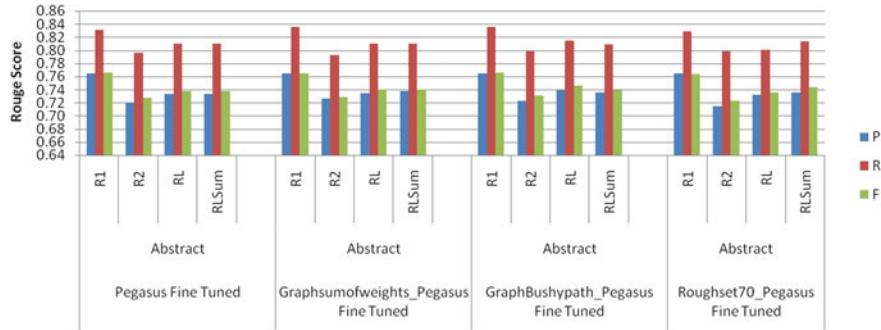
**Fig. 4** Rouge scores of hybrid methods (graph with sum of weights and graph with bushy path methods combined with BART and Pegasus fine-tuned models)



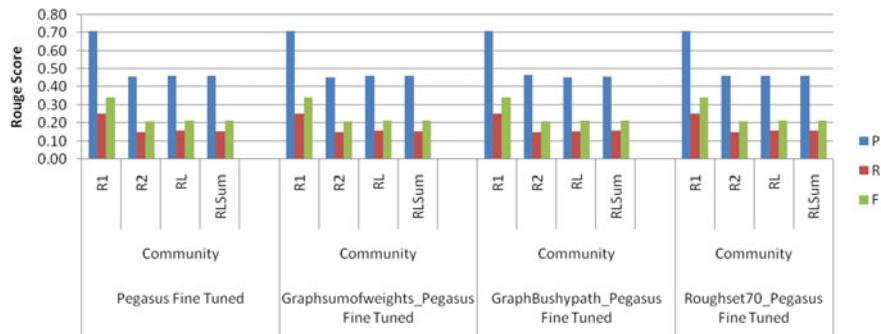
**Fig. 5** Rouge scores of hybrid methods (rough set TFIDF method with BART, Pegasus and BigBird models)

abstractive method alone. From the above experiments, we can conclude that the best rouge scores are measured when the extractive summary produced by the rough set TFIDF graph method is input to the Pegasus fine-tuned model for generating the abstractive summary.

**Rouge Scores of Pegasus Fine-Tuned Model and Hybrid Methods with Pegasus over Abstract, Community and Human Summary** The ScisummNet corpus has research articles in ACL computational linguistics and the NLP domain and three output summaries each: abstract, community summary and human summary. This corpus was built as part of the CL-SciSumm Shared Task conducted in 2018, 2019 and 2020. A random sample of 10 articles is used in the performance analysis here. Rouge scores of the summary generated by the abstractive Pegasus fine-tuned model and the hybrid graph and rough set TFIDF methods combined with it against the abstract, community and human summary are discussed in this section. Figure 6



**Fig. 6** Rouge scores of Pegasus fine-tuned model and hybrid methods with Pegasus fine-tuned model over abstract

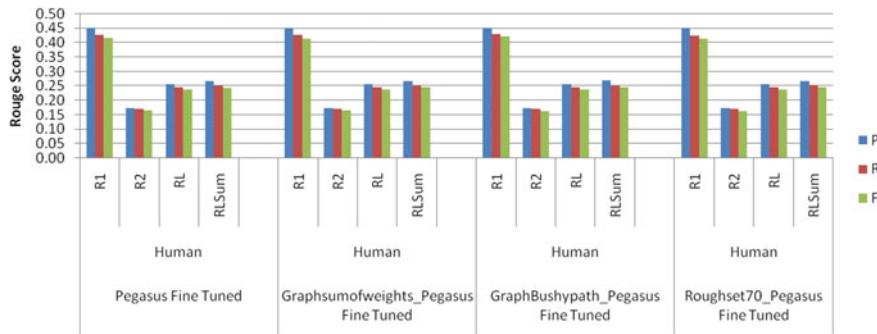


**Fig. 7** Rouge scores of Pegasus fine-tuned model and hybrid methods with Pegasus fine-tuned model over community summary

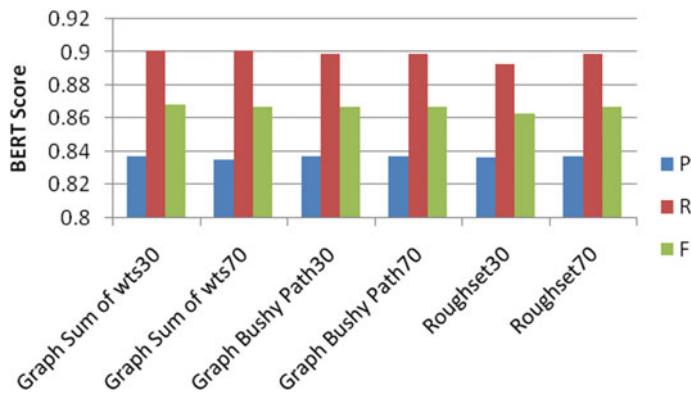
shows the rouge scores of the Pegasus fine-tuned model, graph with the sum of weights and graph with bushy path method combined with it against the abstract of the paper. Figure 7 shows the rouge scores against the community summary, and Fig. 8 shows the scores against the human summary.

## 5.2 BERT Score for Similarity Score

BERT score is an automatic evaluation metric computing the similarity score for each token in the generated summary with the gold summary. It uses the pre-trained contextual embedding from the BERT model and calculates the cosine similarity. The precision, recall and f-measure is calculated [4]. For the default model for the English language, the Roberta large is used in this implementation. Figure 9 shows the BERT scores of different extractive methods used in this work. The graph with the sum of weights and the graph with the bushy path do not have a significant difference

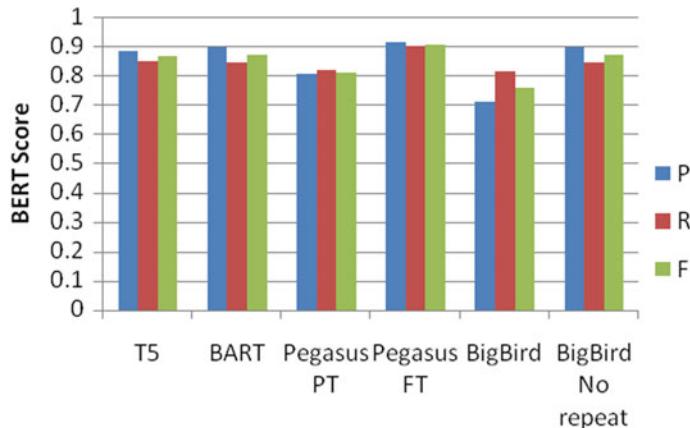


**Fig. 8** Rouge scores of Pegasus fine-tuned model and hybrid methods with Pegasus fine-tuned model over human summary

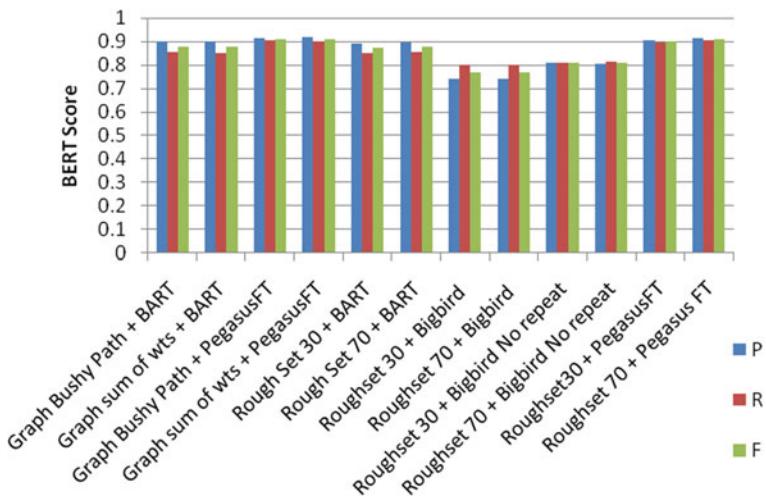


**Fig. 9** BERT score of extractive summarization methods

between BERT scores of the summary of 30 and 70% length. The rough set TFIDF method has slight improvement in BERT scores for the summary of 70% over 30% length. Figure 10 shows the BERT scores of different abstractive methods used in this work. The Pegasus fine-tuned model has the best scores compared to T5, BART, Pegasus pre-trained, BigBird and BigBird no\_repeat models. Figure 11 shows the BERT scores of different hybrid methods used in this work. The best values of the BERT score are measured when the extractive summary generated by the rough set, graph with the sum of weights and graph with bushy path methods is given as input to the Pegasus fine-tuned model.



**Fig. 10** BERT score of abstractive summarization models



**Fig. 11** BERT score of hybrid summarization methods

## 6 Conclusion and Future Scope

In this research, a comprehensive analysis of various extractive, abstractive and hybrid methods for journal article summarization is conducted. For the extractive summarization, three distinct methods are employed: the graph with the sum of weights, the graph with the bushy path and the rough set TFIDF methods. For abstractive summarization, cutting-edge pre-trained models like BART, Pegasus, T5 and BigBirdPegasus were used. The Pegasus pre-trained model was fine-tuned using the ScisummNet dataset, which comprises over 1000 articles related to sci-

tific literature. To further enhance the performance, a hybrid approach was proposed and implemented. This hybrid method combined the strengths of both extractive and abstractive methods. The outputs of the graph-based and rough set TFIDF extractive models were fed into the Pegasus fine-tuned model, leading to even more promising results. The hybrid models showed significant improvements in terms of rouge and BERT scores, surpassing the individual extractive and abstractive methods. The best rouge and BERT scores were obtained for the Pegasus fine-tuned and BART abstractive models and for the hybrid models proposed in this work combining the graph and rough set TFIDF methods with Pegasus fine-tuned models in the current settings. In the future, there is a tremendous potential to expand the scope of this work beyond journal articles and include a wide variety of document types. Furthermore, we envision breaking language barriers and enabling the analysis of documents in various regional languages. The work can be extended to documents other than journal articles and documents of regional languages.

## References

1. KSK, Mathew S (2020) Survey of scientific document summarization techniques. *Comput Sci* 21(2)
2. Chandrasekaran M, Yasunaga K, Radev M, Freitag D, Kan M (2019) Overview and results: CL-SciSumm shared task
3. Chin-Yew L (2004) ROUGE: a package for automatic evaluation of summaries. In: Text summarization branches out, ACL, pp 74–81
4. Zhang T, Kishore V, Wu F, Weinberger Q, Artzi Y (2019) BERTScore: evaluating text generation with BERT
5. KSK, Mathew S (2023) High impact of rough set and KMeans clustering methods in extractive summarization of journal articles. *J Inf Sci Eng* 39(3):561–574
6. Mihalcea R, Tarau P (2004) TextRank: bringing order into text. In: Proceedings of the conference on EMNLP. ACL, pp 404–411
7. Erkan G, Radev DR (2004) LexRank: graph-based lexical centrality as salience in text summarization. *J Artif Intell Res* 22:457–479
8. Mandar M, Amit S, Chris B (1997) Automatic text summarization by paragraph extraction. In: Workshop on intelligent and scalable text summarization
9. Krishna MRVV, Reddy S (2015) Extractive summarization technique based on fuzzy membership calculation using rough sets. *IJCAI* 14(2)
10. Rush AM, Chopra S, Weston J (2015) A neural attention model for abstractive sentence summarization. <https://doi.org/10.48550/ARXIV.1509.00685>
11. Chopra S, Auli M, Rush AM (2016) Abstractive sentence summarization with attentive recurrent neural networks. In: Proceeding of the 2016 ACL: HLT, pp 93–98
12. Tan J, Wan X, Xia J (2017) Abstractive document summarization with a graph-based attentional neural mode. Proceeding of the 55th ACL, vol 1. pp 1171–1181
13. Danqing W, Pengfei L, Yining Z, Xipeng Q, Xuanjing H (2020) Heterogeneous graph neural networks for extractive document summarization. In: Proceeding of the 58th ACL, pp 6209–6219
14. Jacob D, Ming-Wei C, Kenton L, Kristina L (2018) BERT: pre-training of deep bidirectional transformers for language understanding, pp 4171–4186
15. Mike L, Yinhan L, Naman G, Marjan G, Abdelrahman M, Omer L, Ves S, Luke Z (2019) BART: denoising sequence-to-sequence pre-training for natural language generation. Translation, and Comprehension

16. Jingqing Z, Yao Z, Mohammad S, Peter L (2019) PEGASUS: pre-training with extracted gap-sentences for abstractive summarization
17. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Peter L (2019) Exploring the Limits of transfer learning with a unified text-to-text transformer. *J Mach Learn Res*
18. Abigail S, Liu PJ, Christopher DM (2017) Get to the point: summarization with pointer-generator networks. <https://doi.org/10.48550/ARXIV.1704.04368>
19. Esin D, He H, Mona D (2020) FEQA: a question answering evaluation framework for faithfulness Assessment in abstractive summarization. In: Proceeding of the 58th ACL, pp 5055–5070
20. Iz B, Matthew, Arman PC (2020) Longformer: the long-document transformer. <https://doi.org/10.48550/ARXIV.2004.05150>
21. Zaheer M, Guruganesh G, Dubey A, Ainslie J, Alberti C, Ontanon S, Pham P, Ravula A, Wang Q, Yang L, Ahmed A (2020) Big bird: transformers for longer sequences. <https://doi.org/10.48550/ARXIV.2007.14062>
22. Tobias R, Xiaoxia W, Yinhan L (2021) Hierarchical learning for generation with long source sequences. <https://doi.org/10.48550/ARXIV.2104.07545>
23. Sandeep S, Raymond L, Jonathan P, Christopher P (2019) On extractive and abstractive neural document summarization with transformer language models. <https://doi.org/10.48550/ARXIV.1909.03186>
24. Bajaj A, Dangati P, Krishna K, Ashok Kumar P, Uppaal R, Windsor B, Brenner E, Dotterrer D, Das R, McCallum A (2022) Long document summarization in a low resource setting using pretrained language models. In: Proceedings of the 59th ACL, pp 71–80. <https://doi.org/10.18653/v1/2021.acl-srw.7>
25. Cachola I, Lo K, Cohan A, Weld DS (2020) TLDR: extreme summarization of scientific documents. EMNLP, Findings of the ACL
26. Thijss (2017) Improving the compositionality of word embeddings. In: Proceedings of the web conference, pp 1083–1093. <https://doi.org/10.1145/3178876.3186007>
27. Vivek G, Preerna B, Pegah N, Harish K (2021) SumPubMed: summarization dataset of PubMed scientific articles. In: Proceeding of the 59th ACL, pp 292–303
28. Cohan A, Dernoncourt F, Kim DS, Bui T, Kim S, Chang W, Goharian N (2018) A discourse-aware attention model for abstractive summarization of long documents. In: Proceeding of the ACL: HLT, pp 615–621
29. Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, Davison J, Shleifer S, Platen PV, Ma C, Jernite Y, Plu J, Xu C, Scao TL, Gugger S, Drame M, Lhoest Q, Rush AM (2020) HuggingFace’s transformers: state-of-the-art natural language processing
30. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Jeffrey Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D (2020) Language models are few-shot learners
31. Ibrahim AN, Bachir MME (2022) Automatic summarization of scientific articles: a survey. *JKSUCI* 34(4):1011–1028. <https://doi.org/10.1016/j.jksuci.2020.04.020>

# Differential Evolution Wrapper-Based Feature Selection Method for Stroke Prediction



Santwana Gudadhe and Anuradha Thakare

**Abstract** Stroke is a medical condition in which a blood vessel in the brain bursts and causes brain damage. After realizing the wide-ranging effects, a brain infarction can have on a community, significant efforts have been made to enhance stroke therapy and diagnosis. The medical practitioner can benefit from more precise diagnosis if the occurrence of strokes can be predicted using patient medical records. Identifying the important features decreases the number of features by eliminating irrelevant or misleading, noisy and redundant data which can accelerate the process of prediction. The proposed work, involves the wrapper-based differential evolution approach for best feature selection and based on best features performing the stroke prediction using random forest algorithm. To evaluate the performance of proposed work, wrapper-based sequential forward, backward, and wrapper-based built around random forest algorithm are evaluated. Using a wrapper-based feature selection method, this work identifies critical characteristics, and then proposes stroke prediction accuracy estimation. The following are the findings from our investigation into methods for selecting features that make use of proposed wrapper-based differential evolution algorithm. The most significant features are age, ever married, residence type, and Avg\_glucose\_level. The proposed prediction model achieved maximum accuracy of 95.79%.

**Keywords** Wrapper-based differential evolution · Feature selection · Random forest algorithm · Stroke prediction

---

S. Gudadhe (✉) · A. Thakare  
Pimpri Chinchwad College of Engineering, Pune, India  
e-mail: [s.santwana20@gmail.com](mailto:s.santwana20@gmail.com)

A. Thakare  
e-mail: [anuradha.thakare@pccoepune.org](mailto:anuradha.thakare@pccoepune.org)

## 1 Introduction

Among the leading causes of death in the world, stroke ranks high. Stroke risk assessment is important but difficult because there are so many potential contributing variables. In 2019, cardiovascular diseases were responsible for 32% of all deaths worldwide [1]; with 17.9 million individuals dying as a result (85% were due to stroke and heart attack). Strokes are more prevalent in the elderly and can cause a wide range of symptoms including hemiplegia, slurred speech, and loss of consciousness, as well as other forms of brain damage. Adults can become severely disabled and even die, as a result of them. Recognizing or predicting an impending stroke in its early stages may make it possible to substantially mitigate its effects. Numerous studies and clinical trials have pinpointed a number of variables that increase one's likelihood of suffering a stroke.

In addition, stroke develops quickly and presents with a wide range of symptoms. The onset of symptoms can range from imperceptibly sluggish to alarmingly rapid. One can even experience sign while sleeping and awake with them. The sudden onset of any of these signs indicates a stroke. Common symptoms include weakness or paralysis in the arms or legs (typically on one side of the body), numbness in the arms, legs, or sometimes the face, difficulty speaking or walking, dizziness, decreased eyesight, headache, vomiting, and a downward turn in the angle of the lips. Consciousness is lost and a coma sets in as a final result of serious strokes [2, 3].

However, research into isolating the most crucial variables required for stroke prediction is lacking. Previous research in this field [4–8] shows that the end performance of a machine learning framework is affected by the features that are prioritized. Rather than using all of the features in the feature space, it is crucial to obtain the optimal mix of features. Prior to applying a classification method, it is recommended that redundant attributes and/or attributes that are completely unrelated to a class be recognized and removed. That's why it's crucial for healthcare data miners to understand the interconnectedness of the risk factors recorded in EHRs and how those factors affect the reliability of stroke forecasts on their own. Examine the patient's electronic health record (EHR) for the most relevant information that can be used to forecast a stroke.

This work mainly includes identifying the important features from existing feature set and based on best optimal features perform the brain stroke prediction using machine learning algorithms. The contributions of this paper are: wrapper-based differential evolution approach to select the best features and stroke prediction using random forest algorithm is proposed and a faster convergence rate is proposed, which benefits both local and global search and yields near-optimal solutions, which in turn enhances the accuracy of stroke prediction. The article is structured as follows. This chapter two is a summary of relevant research. The material and methods are covered in Sect. 3. Subsections elaborate on the dataset employed, proposed technique, Sect. 4 presents a discussion of the findings. Sections 5 and 6 conclude with certain analysis and final thoughts.

## 2 Literature Review

The most critical and significant characteristics of stroke patients must be chosen for accurate risk detection. Filter-based, wrapper-based, and embedded approaches are the primary feature selection techniques. Specifically, this work focuses on the related work of feature selection using wrapper-based differential evolution approach, as well as other related algorithms and brain stroke prediction using machine learning approach.

Mutual information, ranking, and weighting-based techniques make up the filter feature selection approach. In order to assess the usefulness of features in predicting class labels and the overlap between candidates, mutual information-based filter techniques use this information to make their decisions. Wrapper procedures employ a classifier to determine the efficacy of a feature set and then use that information to find the best possible subgroup of features within that space. Wrapper techniques may be used to discover useful feature subsets [9] for a given classifier.

Many feature subset selection methods have been proposed in the literature. In some research, feature selection was accomplished with the help of differential evolution (DE). In [10], a wrapper-based feature selection algorithm for the prediction of the bank marketing problem was proposed. This algorithm consists of an artificial neural network and a self-adaptive differential evolution optimization algorithm. In [11], the authors discuss the various discretization methods that can be incorporated into differential evolution and how they affect the quality of the obtained feature subsets. In [12], the use of binary differential evolution based on individual entropy to optimize feature subsets was suggested. When compared with GA, ACO, PSO, and DE, the outcomes for the binary differential evolution algorithm were superior in terms of the size of the feature subset and the amount of time it took to achieve. In [13], the authors propose a differential evaluation algorithm for intrusion detection systems that is based on a wrapper and uses a classifier developed using extreme learning machines. The proposed method successfully optimized the number of features being processed by the system while also boosting accuracy, decreasing false alarm rates and speeding up processing. In [14], an innovative multi-variant differential evolution (MVDE) algorithm was suggested to address 15 prominent real-world problems culled from the UCI repository. Research work in the area of brain stroke prediction uses the concept of optimal feature selection and prediction using machine learning algorithms.

Brain stroke prediction was performed using stacking approach in [15]. Feature ranking was used to obtain the important feature and finally machine learning classification algorithms were used to perform the prediction. Pipeline-based approach to select optimal features and prediction using machine learning classifier was proposed in [16, 17]. Brain stroke occurrences prediction using deep learning and machine learning approach was experimented in [18]. Based on physiological parameters and machine learning algorithms, brain stroke prediction was performed. Voting-based classifier was experimented [19]. Analysis of various features and brain stroke prediction at early using machine learning algorithm was experimented in [20].

### 3 Materials and Methods

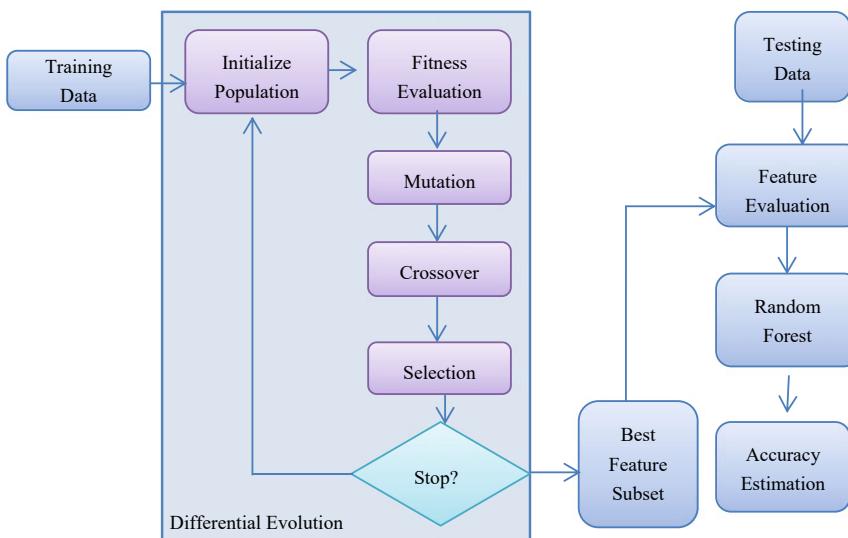
The dataset, the suggested system architecture, the machine learning models, the experiments, and the evaluation matrices are all described here.

#### 3.1 Dataset

Kaggle [21] gathered experimental data, as 5110 observations. This dataset links stroke risk to gender, age, numerous diseases, and smoking. Most EHR data are for non-stroke cases, skewing stroke incidence. Dataset includes patient ID, gender, job type, age, residence type, heart disease, ever married, hypertension, avg\_glucose level, smoking status, bmi, and stroke.

#### 3.2 Proposed System

Proposed work includes the following steps: Fig. 1 shows the proposed system architecture. Dataset used for experimentation is preprocessed and passed to wrapper-based differential evolution algorithm to obtain best feature set. Finally, best feature set is used to perform the prediction using random forest, XGBoost, and gradient boost algorithms. All steps are discussed in detail below.



**Fig. 1** Proposed system architecture

## Data Preprocessing

The elimination of data anomalies and noise is a part of the data preprocessing step. A class balance can be achieved through the use of resampling, along with the elimination of null values and redundant data. SMOTE [22] attempts to bridge the gap that exists between participants who survived a stroke and those who did not survive the event. For fairness, the minority group, or “stroke,” was oversampled. There was neither dropping nor any kind of data imputation necessary because there were no missing or null numbers. After the information has been preprocessed, it is split into a training set of 70% and a testing set of 30%. A wrapper-based differential evolution algorithm is suggested as a means of achieving the goal of obtaining the best possible feature set.

## Wrapper-Based Differential Evolution Algorithm (WBDEA)

Once the training set has been partitioned, the evolutionary process can be applied to pick the most useful features. The proposed differential evolution algorithm is built on a wrapper and consists of the following operations: this includes seeding, assessing viability, mutation, crossover, and selection.

A random and information gain score is used to seed the community. Mutation and crossing operators are used to produce an alternative solution  $S_i$  for each solution  $V_i$ . Selection of  $S_i$  over  $V_i$  occurs if, according to the fitness function,  $S_i$  is the superior option. Following the selection phase, the best answer (denoted sBest) in the population is identified. Using this value as a guide, any unnecessary features from the subset are removed, and any useful features from the unselected feature set are added. Training and testing groups undergo dimensionality reduction based on the features that were chosen during the evolutionary process. The forecast model is developed using the smaller training set, and its accuracy is checked employing the smaller test set.

## Seeding

Two distinct methods of population seeding are employed in the proposed study.

**Algorithmic shuffle:** As a potential feature subset, each solution is seeded with a different permutation of the features. As a first step, we determine the information gain scores for each characteristic along with its corresponding class labels. Once all characteristics have been collected, they are ranked from most important to least. At last, for each answer, a small subset of the most important features is chosen.

## Fitness Function

Minimizing the number of features used to make a prediction and optimizing the accuracy of those predictions are two criteria used to assess the proposed feature selection technique. Taking into account the minimization issue, we can define the fitness function to be a weighted linear aggregation function as shown in Eq. 1.

$$\text{Fitness} = \alpha * \text{Error Rate} + (1 - \alpha) * \frac{X}{Y} \quad (1)$$

where  $X$  is the number of features in the feature subset that was chosen, and  $Y$  is the overall number of features present in the data, error rate is the fraction of instances that was incorrectly classified. The K-nearest-neighbors method is taken into account as a classification option when determining the error rate. When comparing two examples with different types of an attribute, Manhattan distance is used as the distance metric. The fivefold cross validation technique first reduces the training set to a smaller size by using a feature subset selection, and then divides that smaller set into five separate sets in order to assess the error rate of a potential solution. If a randomly generated number between 0 and 1 is less than a user-specified value, then the elimination process is carried out; otherwise, the add process is carried out, allowing for a probabilistic search for the best answer in the population to discover fitter solutions. After the elimination and addition procedures have been carried out in a probabilistic fashion, the current sBest and the updated sBest' solutions are compared and a winner is determined. When comparing two solutions, fitness is taken into account and the one with a higher number is maintained as the best option. The process is repeated until the desired number of trials has been reached (stopping criteria).

### Prediction Model

Once the best sets of features are obtained by using wrapper-based differential evolution algorithm, these features are passed to random forest to perform the prediction. Finally, prediction accuracy is estimated. Along with prediction accuracy, precision, recall,  $F$ \_Score measure is evaluated. In order to evaluate the performance of prediction model, wrapper-based built around random forest (BORUTA) is also experimented, where best sets of features are obtained using BORUTA and obtained best set of features are passed to random forest classifier. Result section discusses the accuracy estimation using random forest algorithm. To perform the comparative analysis, gradient boost and XGBoost are also experimented and discussed in below section.

## 4 Result Analysis

The experimental findings using the suggested differential evolution algorithm with a wrapper are discussed here. Because the classifier used can have an effect on the final feature selection product, this research used KNN as the classifier for all algorithms with  $K = 5$ . The distance measure used was the Manhattan distance. Table 1 specifies the range of allowed parameter values for the method. It includes crossover rate, scaling factor,  $k$ -value in KNN, no. of chromosomes, and maximum number of generations set for proposed architecture. The stroke prediction in the proposed wrapper-based differential evolution prediction model is handled by the random forest method. Early on, information gain is used to assess the features in order to determine the crucial ones. Some of the most notable characteristics

**Table 1** Parameter value set for WBDEA

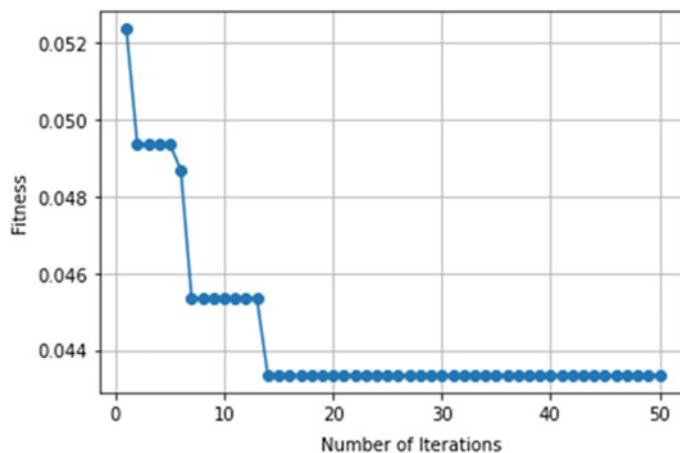
Parameters	Values
CR (crossover rate)	0.8
F (scaling factor)	0.5
K ( $k$ -value in KNN)	5
N (number of chromosomes)	10
T (maximum number of generations)	50

noticed are the subjects' ages, ever\_married, body mass indexes, Avg\_glucose\_level, Residence\_type, and occupations.

The random forest method is used to judge how well the strategy works. There was a peak in the proposed model's forecast accuracy of 95.79%. As can be seen in Fig. 2, the suggested method also demonstrates respectable performance in terms of convergence rate generated during experimentation, as it reaches the global minimum in a reasonable amount of time.

Wrapper-based sequential forward selection [23], wrapper-based sequential backward elimination, and wrapper-based built around random forest algorithm (BORUTA) [24] have been compared and evaluated to gauge the efficacy of the proposed wrapper-based differential evaluation-based stroke prediction method. Table 2 displays the results of a comparison between the suggested model and the random forest algorithm, another wrapper-based feature selection algorithm.

Wrapper-based sequential forward selection, backward elimination, and built around random forest algorithm are evaluated using random forest, XGBoost, and gradient boost algorithm. Based on the result analysis, it is observed that, wrapper-based sequential forward selection using XGBoost algorithm achieved maximum accuracy of 91.60%. For sequential backward elimination, maximum accuracy

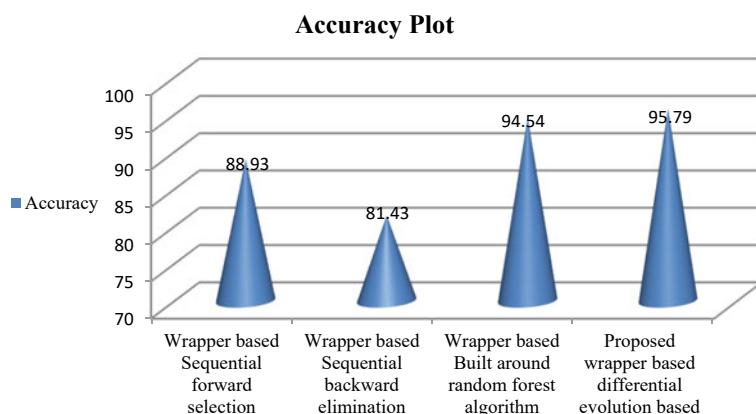


**Fig. 2** Convergence rate for proposed WBDEA

**Table 2** Comparative analysis of proposed WBDEA

Methods	Algorithm	Accuracy	Precision	Recall	F_Score
Wrapper based sequential forward selection	Random forest	88.93	95.41	92.78	94.08
	Gradient boost	79.40	96.06	81.63	88.25
	XGBoost	91.60	95.38	95.79	95.58
Wrapper based sequential backward elimination	Random forest	81.43	94.91	84.97	89.67
	Gradient boost	79.40	96.06	81.63	88.25
	XGBoost	54.40	95.48	54.50	69.39
Wrapper based built around random forest algorithm	Random forest	94.54	94.85	99.65	97.19
	Gradient boost	93.89	94.81	98.96	96.85
	XGBoost	94.86	94.86	100	97.36
Proposed system	Random forest	95.79	95.79	100	97.81
	Gradient boost	93.86	94.89	98.65	96.81
	XGBoost	94.9	94.81	99.65	97.19

observed for random forest is 81.43%. When compared with forward and backward selection, wrapper-based built around random forest algorithm shows improved accuracy for random forest is 94.54%. Finally, wrapper-based differential evolution-based approach is evaluated for random forest, XGBoost, and gradient boost algorithms and maximum accuracy achieved for random forest is 95.79%. Figure 3 shows the comparative analysis of accuracy plot for proposed WBDEA with wrapper-based sequential forward, backward elimination, and wrapper-based built around random forest algorithm methods using random forest algorithm.

**Fig. 3** Comparative analysis of accuracy plot for proposed WBDEA with another feature selection methods using random forest algorithm

## 5 Discussion

The proposed work includes wrapper-based differential evolution for best feature selection and stroke prediction using random forest algorithm. Analyzing the important features can be the key idea to predict the stroke occurrence in a more accurate way. To identify the important wrapper-based differential evolution, best feature selection model is proposed. The evaluated features using proposed methods are age, ever married, Avg\_glucose\_level, and residence\_type. Once the best feature is obtained, it is passed to random forest, XGBoost, and gradient boost algorithms to perform the stroke prediction. The proposed work achieved better performance in terms of prediction accuracy. In order to check the performance of proposed work, wrapper-based sequential forward, backward, and wrapper-based built around random forest algorithm are experimented. From result analysis, sequential forward selection achieved maximum accuracy of 88.93% using random forest algorithm. Sequential backward elimination achieved maximum accuracy of 81.43%, wrapper-based built around random forest achieved maximum accuracy of 94.54% using random forest algorithm. In all prediction algorithms, random forest performs well when compared with other algorithms. Proposed work is evaluated using random forest algorithm and achieved highest accuracy of 95.79%.

## 6 Conclusion

Analyzing the important features can be the key idea to predict the stroke occurrence in a more accurate way. Proposed work includes the best feature selection using wrapper-based differential evolution approach and based on best feature selection, stroke prediction is performed using random forest algorithm. In order to evaluate the performance of proposed work, wrapper-based sequential forward, backward, and wrapper-based built around random forest algorithm are experimented and prediction accuracy observed. In the proposed work, wrapper-based differential evolution-based feature selection approach (WBDEA) with random forest achieved the maximum prediction accuracy of 95.79% in all approaches. Considering all performances comparison, random forest performs well for all wrapper-based feature selection methods. The identified important features are age, ever\_married, Avg\_glucose\_level, and residence\_type.

## References

1. [https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)). Last accessed 12 Feb 2023
2. Mosley I, Nicol M, Donnan G, Patrick I (2007) Stroke symptoms and the decision to call for an ambulance. *Stroke* 38(2):361–366
3. Lecouturier J, Murtagh MJ, Thomson RG, Ford GA, White M, Eccles M, Rodgers H (2010) Response to symptoms of stroke in the UK: a systematic review. *BMC Health Serv Res* 10:1–9

4. Illhoi Yoo, Patricia Alafaireet, Miroslav Marinov, Keila Pena-Hernandez, Rajitha Gopidi, Jia-Fu hang & Lei Hua (2012) Data mining in healthcare and biomedicine: a survey of the literature. *J Med Syst* 36(4):2431–2448
5. García S, Luengo J, Herrera F (2018) Tutorial on practical tips of the most influential data preprocessing algorithms in data mining. *Knowl Based Syst* 98:1–29
6. Gudadhe S, Thakare A, Anter AM (2023) A novel machine learning-based feature extraction method for classifying intracranial hemorrhage computed tomography images. *Healthc Analytics* 3:100196. ISSN 2772-4425
7. Gudadhe SS, Thakare AD, Oliva D (2023) Classification of intracranial hemorrhage CT images based on texture analysis using ensemble-based machine learning algorithms: a comparative study. *Biomed Sig Proc Control* 84:104832. ISSN 1746-8094
8. Gudadhe S, Thakare A (2023) Multivariate analysis of Ischaemic lesions using computed tomography and CT perfusion imaging: critical review. *Comput Methods in Biomed Eng Imaging Visual Taylor Francis* 16:2168–1163
9. Chyžhýk D, Savio A, Grana M (2014) Evolutionary ELM wrapper feature selection for Alzheimer's disease CAD on anatomical brain MRI. *J Neurocomputing* 128:73–80
10. Fister D, Fister I, Jagric T, Fister I Jr, Brest J (2019) Wrapper-Based feature selection using self-adaptive differential evolution. *Commun Comput Inf Sci* 1092
11. Zorić B, Bajer D, Dudjak M (2020) Wrapper-based feature selection via differential evolution: benchmarking different discretisation techniques. In: International conference on smart systems and technologies, pp 89–96
12. Tao Li; Hongbin Dong; Jing Sun (2019) Binary differential evolution based on individual entropy for feature subset optimization. *IEEE Access* 7:24109–24121
13. Wathiq Laftah Al-Yaseen, Ali Kadhum Idrees, Faezah Hamad Almasoudy (2022) Wrapper feature selection method based differential evolution and extreme learning machine for intrusion detection system. *Pattern Recognit* 132
14. Hassan S, Hemeida AM, Alkhalfaf S, Mohamed AA, Senju T (2020) Multi-variant differential evolution algorithm for feature selection. *Sci Rep* 10:17261
15. Dritsas E, Trigka M (2022) Stroke risk prediction with machine learning techniques. *Sensors (Basel)* 13:4670
16. Gudadhe, S., Thakare, A., Predictive Analytics for Stroke Prediction Using a Wrapper-Based Feature Selection Pipeline Approach in Machine Learning. In: Chaki, N., Roy, N.D., Debnath, P., Saeed, K. (eds) Proceedings of International Conference on Data Analytics and Insights, ICDAI 2023. Lecture Notes in Networks and Systems, vol 727. Springer, Singapore. [https://doi.org/10.1007/978-981-99-3878-0\\_25](https://doi.org/10.1007/978-981-99-3878-0_25) (2023).
17. S. Gupta and S. Raheja, Stroke Prediction using Machine Learning Methods, 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, pp. 553–558 (2022).
18. Rahman S, Sarkar A (2022) Prediction of Brain Stroke using Machine Learning Algorithms and Deep Neural Network Techniques. *European Journal of Electrical Engineering and Computer Science.* 7:23–30
19. Tahia Tazin, Md Nur Alam, Nahian Nakiba Dola, Mohammad Sajibul Bari, Sami Bourouis, and Mohammad Moniruzzaman Khan, Stroke Disease Detection and Prediction Using Robust Learning Approaches, *Journal of Healthcare Engineering*, Vol. 2021, Article ID 7633381 (2021).
20. Padimi V, Telu VS, Ningombam DD (2022) Performance analysis and comparison of various machine learning algorithms for early stroke prediction. *ETRI J* 1–15
21. <https://www.Kaggle.com/fedesoriano/stroke-prediction-dataset..Strokepredictiondataset>. Last accessed 20 Jan 2023
22. Maldonado S, Lopez J, Vairetti C (2019) An alternative SMOTE oversampling strategy for high-dimensional datasets. *Appl Soft Comput J* 76:380–389
23. [http://rasbt.github.io/mlxtend/user\\_guide/feature\\_selection/SequentialFeatureSelector](http://rasbt.github.io/mlxtend/user_guide/feature_selection/SequentialFeatureSelector). Last accessed 20 Jan 2023
24. Kursa MB, Rudnicki WR (2010) Feature selection with the Boruta package. *J Stat Softw* 36(11):1–13

# Parkinson's Disease Identification from Speech Signals Using Machine Learning Models



Rahul Saxena and J. Andrew 

**Abstract** Parkinson's disease (PD) is a common chronic neurodegenerative illness characterised by continuous nervous system degradation. This condition is more prevalent in the elderly. In Parkinson's, dopaminergic neurons die at an early stage, resulting in a progressive neurodegenerative condition. PD can cause a various symptom of non-motor and motor, including smell and speech. One of the problems that patients with Parkinson's may face is a pronunciation or having difficulty while speaking. As a result, early diagnosis is critical in minimising the potential effects of disease-related speech disorders. This journal intends to build a categorisation scheme for Parkinson's disease to distinguish between healthy individuals and PD sufferers and create a hybrid classifier by combining distinct machine learning models. For this journal, we have implemented Support Vector Machine (SVM), K-Nearest Neighbour (KNN), Random Forest classifier, and Logistic Regression ML techniques and acquired the classification report. The results showed that Random Forest has outperformed other ML techniques with 89.47% accuracy for the testing set.

**Keywords** Machine learning · Parkinson's disease · Random Forest · Feature selection · Classification · Speech features

---

R. Saxena

Department of Instrumentation and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka 576104, India  
e-mail: [rahul.saxena@learner.manipal.edu](mailto:rahul.saxena@learner.manipal.edu)

J. Andrew 

Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka 576104, India  
e-mail: [andrew.j@manipal.edu](mailto:andrew.j@manipal.edu)

## 1 Introduction

PD is a neurodegenerative condition that affects the twain motor and non-motor capabilities. It results in motor symptoms, example tremors, stiffness, bradykinesia, and involuntary movements due to the deprivation of dopamine-producing neurons in the brain. Furthermore, PD can cause non-motor symptoms like depression, nervousness, mental impairment, sleep issues, and autonomic dysfunction. These signs can show up before or after motor symptoms emerge and have no direct connection to dopamine depletion. Non-motor symptoms frequently go undiagnosed and untreated, yet they can have a big influence on a person's well-being. To enhance overall functioning and well-being of a person's life, it is necessary to address one and the other non-motor and motor symptoms of PD. While non-pharmacological interventions like psychotherapy, cognitive behavioural therapy, and mindfulness-based therapies can help manage non-motor symptoms, medications that raise dopamine levels in the brain, physical therapy, exercise, and deep brain stimulation surgery can be used to control motor symptoms. PD is believed to cause by mix of environmental factors and genetics, while the specific origin is unknown.

ANN is used in DL, a branch of machine learning, to model and resolve complicated issues. Layers of interconnected processing nodes or neurons make up neural networks, which learn from data and extract features to produce classifications or predictions. Deep learning algorithms have achieved outstanding results in various domains, including computer vision, natural language processing, and speech recognition. In a study by Tsanas et al. [1], PD was discovered utilising machine learning methods from voice recordings. A total of 22 variables, comprising various speech-related traits including pitch, jitter, and shimmer, were retrieved from each voice recording by the researchers from the 42 PD patients and the 42 healthy control participants. These characteristics were then utilised to train a number of machines learning models, including support vector machines (SVMs) and decision trees, to categorise the recordings as either PD or healthy controls. The proposed model yielded an accuracy of 92.7% and an area under the curve (AUC) of 0.98, the best model for differentiating between PD and healthy participants based on voice recordings was an SVM. In order to detect PD utilising non-invasive and readily available speech recordings, this study shows the promise of machine learning. This could result in an earlier diagnosis and improved disease management. Deep learning and neural networks are well-liked techniques for resolving complicated issues as a result of their success in a variety of fields.

The objective of this research paper are:

- A classification scheme is being developed to identify between healthy people and people with Parkinson's disease.
- Creating a hybrid classifier by combining different machine learning models with neural networks (NN) to achieve the best accuracy possible.

PD is a condition that might diagnosed and treated early. With the use of ML algorithms, clinicians may capable of make earlier, more accurate diagnoses of Parkinson's disease, which could improve patient outcomes.

The rest of this paper is arranged as follows: Sect. 2 discusses the background theory and literature review, Sect. 3 discusses methodology and techniques used, Sect. 4 touches on the results obtained from the model, Sect. 5 discusses the conclusion and summarises the paper.

## 2 Literature Review

Speech issues are among the most recognised signs of PD. Parkinson's sufferers may find it challenging to properly communicate due to these issues, which can range in severity from mild to severe. The most popular form of treatment for speech issues is speech therapy, voice therapy, medication, and surgery. Throughout their illness, between 75 and 90% of persons with Parkinson's disease will experience speech issues. This segment discusses the studies that have dated to predict PD using a dataset of speech signals or similar machine learning techniques.

### 2.1 *Machine Learning and Neural Network Techniques for PD Classification*

Quan et al. [2] made a deep learning model technique for the early identification of PD. In this research study, artificial intelligence (AI) was utilised to create a model that extracts time-series dynamic characteristics: two-dimensional convolutional neural networks that are time-distributed and one-dimensional convolutional neural networks that record the relationships between these time series. An accuracy of up to 92% was reached for speech tests using a basic sentence (/loslibros/) and a complicated sentence (/viste/) in Spanish.

Jyotiyana et al. [3] suggested a framework of classification for Parkinson's disease using deep neural networks. The deep neural network-based categorization model was the AI model that was employed. Data from Parkinson's telemonitoring were used. The performance matrices include support, recall, F1-Score, precision, and recall which are utilised. This paper's accuracy rating is 94.87%.

Roobini et al. [4] investigates the categorization of the feature dataset for audio signals in order to identify Parkinson's disease. Regression and XGBoost are two machine learning algorithms that are employed as AI tools in this article. A collection of audio features from the UCI dataset archive is used in this study. The four performance matrices are MCC, sensitivity, accuracy, and specificity. The height accuracy of 96% produced by XGBoost resulted in the MCC statistic (Matthews parametric statistic) of 89%.

Grover et al. [5] suggested deep neural networks' method of predicting the severity of PD. The dataset for this study was provided by the Parkinson's Telemonitoring Voice Dataset from UCI. The following matrices were used as the classification features: Shimmer (dB), Shimmer: APQ3, APQ5, Shimmer: APQ11, Shimmer DDA, NHR, HNR, RPDE, DFA, PPE, Jitter (%), Jitter (Abs), Jitter RAP, PPQ5, DDP. The accuracy is 94.4422% for the train dataset and 62.7335% for the test dataset, respectively.

Tai et al. [6] researched building a model using supervised learning to identify trends in the voice of PD patients. Multilayer Perceptron (MLP), Random Forest (RF), Logistic Regression (LR), and Support Vector Machines (SVMs) are examples of supervised learning techniques, which are employed as AI tools in this work. The dataset was taken from Bot et al. (2016)'s 'Mobile Parkinson Disease Study' originating from the Sage Bionetworks mPower Project. About 88% accuracy was attained [7].

## 2.2 *Hybrid Models with Machine Learning Techniques and Neural Network*

Zhang et al. [8] build two EEG analytic techniques for PD monitoring and recognition. Time-frequency and DL analysis are combined in AI approaches. The dataset consists of clinical sleep EEG data from Shaanxi Provincial People's Hospital. About 99.92% accuracy was attained.

Mehmet Bilal Er [9] uses Mel spectrograms created via Variational Mode Decomposition from denoised audio sources to detect PD from speech sounds utilising long-term memory (LSTM) and pre-trained deep networks. AI techniques either used pre-trained CNNs with LTSM or brand-new CNN designs. The Spanish PC-GITA dataset is utilised in this study. The ResNet-101 + LSTM model achieves the best classification performance, with a VMD of 98.61%.

Uppalapati et al. [10] RF and ANN were combined to build a hybrid neural fuzzy classifier for PD early detection. This work utilised the 'Parkinson's Disease Classification' dataset from the UCI Machine Learning Repository. This paper's accuracy score is 96.23%.

Nilashi et al. [11] provided a fusion method for diagnosing PD using ensemble learning and the capacity to learn from extensive clinical datasets. Two evaluation measures, R2 adjusted (adjusted coefficient of determination) and RMSE (Root Mean Square Error), are used to assess the suggested methodology. The findings show that Expectation–Maximisation + Deep Belief Network + Adaptive Neuro-Fuzzy Inference System (RMSE = 0.537; R2 = 0.893) performs better than other prediction machine learning techniques.

Rahman et al. [12] utilises cutting-edge signal processing algorithms to analyse the various vowel samples obtained from PD patients and healthy participants. ReAlitive SpecTrAI PLP (RASTA-PLP) and Perceptual Linear Prediction (PLP) are two AI

technologies that are extracted from voice signals. After receiving approval from Pakistan's Lady Reading Hospital (Medical Teaching Institution) ethical review board, the database was assembled. The four performance matrices are MCC, sensitivity, accuracy, and specificity. Their algorithm performs 74% accurately, according to the results.

Quan et al. [13] uses a speech signal's time-series dynamic properties to capture PD using a bidirectional long-short-term memory (LSTM) model. CNN and RNN are the two DL models used in this study. The GYENNO SCIENCE Parkinson's Disease Research Centre is where the database was compiled. The findings indicate that to provide reliable predictors and increase the system's adaptability to the input content, a PD detection system would benefit from integrating the Bidirectional LSTM model with dynamic speech features and the end-to-end DL utilising the CNN model.

Shivangi [14] proposed to aid medical professionals and patients in making an early diagnosis of disease, The VGFR Spectrogram Detector and Voice Impairment Classifier are two neural network-based models that have been presented. CNN and ANN are AI tools. The two datasets used were those from the UCI Machine Repository and the PhysioNet Database Bank. The accuracy performance matrices are evaluated and contrasted with those of SVM, XGBoost, and MLP, among other machine learning techniques. Accuracy rates of 89.15% for the Voice Impairment Classifier and 88.1% for the VGFR Spectrogram Detector were attained.

Goyal et al. [15] investigated a hybrid method to extract features from information based on time frequency and resonance. Algorithm using Time–Frequency (T-F) and Resonance-based Sparse Signal Decomposition (RSSD) with CNN is the AI tool employed in this. A 99.37% validation accuracy is attained.

Ma et al. [16] applied deep dual-side learning of PD voice data which is achieved by combining a deep sample learning method with a deep network (deep feature learning). Deep sample learning algorithm (DSL) and group sparse autoencoder (EGSAE) were merged into SVM. In order to test the efficacy of this approach, two datasets were used: the Sakar dataset (Dataset 2) and the LSVT\_voice\_rehabilitation data set (Dataset 1). The suggested algorithm's mean accuracy for the two datasets is 98.4 and 99.6% methodology.

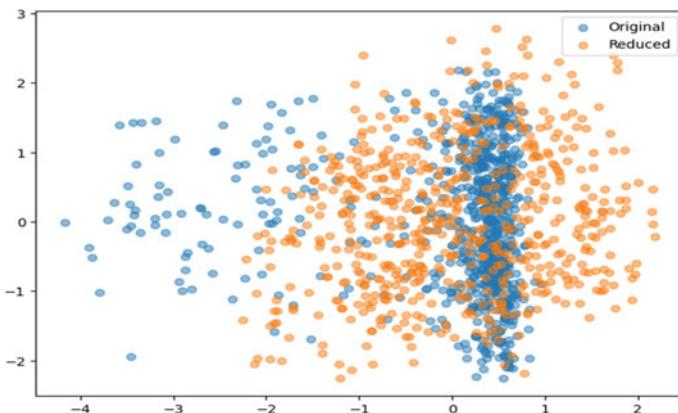
Alshammri's et al. [17] the study uses a variety of machine learning (ML) and deep learning (DL) models, such as Support Vector Machines (SVM), Random Forests (RF), decision trees (DT), K-nearest neighbours (KNN), and Multilayer Perceptrons (MLPs), to identify Parkinson's disease (PD). The dataset consists of 195 voice recordings from 31 patients that were retrieved from the machine learning repository at the University of California, Irvine (UCI). Accuracy, recall, precision, and F1-score were used to evaluate the models, with MLP receiving an exceptional overall accuracy rating of 98.31%, recall of 98%, precision of 100%, and F1-score of 99%.

### 3 System Architecture

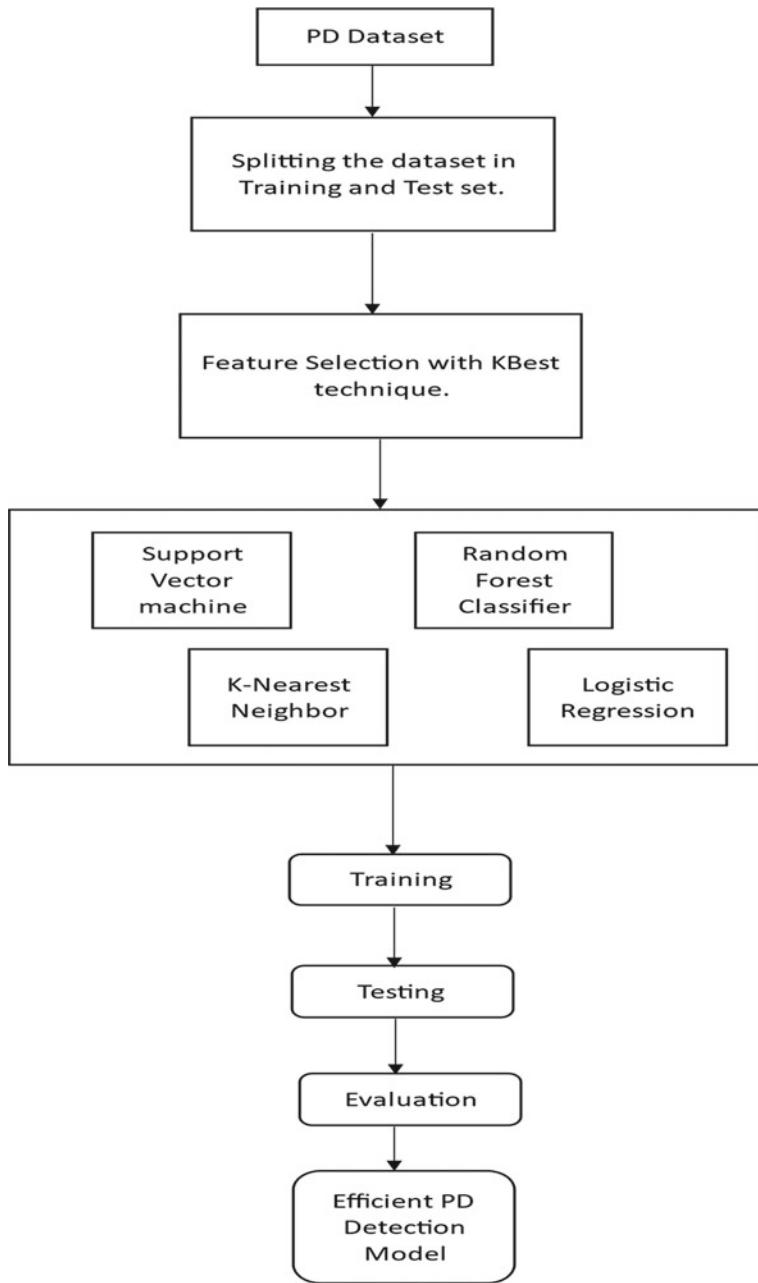
The different stages of a project are shown in Fig. 2. It begins with reading a number of articles in order to comprehend the trends and pinpoint the issue that needs to be resolved. The selection of an appropriate dataset is the next step in the investigation. Table 1 displays a thorough overview of the dataset. The dataset must then be split into training and testing sets as the next stage. This will divide the 755 features into 603 training and 152 testing sets after initially dividing the dataset into training and testing sets, 80/20. Based on their statistical significance, the top K features from a dataset are selected using the feature selection with the KBest approach. Figure 1 shows a scatter plot plotted for the original feature versus the reduced feature selected by the KBest algorithm. It is frequently employed in data analysis and machine learning to reduce the dataset's dimensionality and improve model performance. Complex data can be streamlined, applied, and the results of the chosen feature are then sent to various ML algorithms. This prevents the model from overfitting for the training set, which can have negative effects on accuracy. The output from KBest is then passed to all the ML. After that, the ML model(s) are trained and tested on the dataset divided, and lastly, the best model is selected.

**Table 1** Dataset description

Dataset	Features	Total number of samples		Size of the dataset (MB)
		Parkinson's disease patients	Healthy patients	
Parkinson's disease classification	755	564	192	4.3



**Fig. 1** KBest algorithm implemented on the dataset



**Fig. 2** System architecture

### 3.1 Dataset Details

For this work, the ‘Parkinson’s Disease Classification’ dataset from the UCI Machine Learning Repository was used. The collection includes 192 samples from healthy controls and 564 samples from patients with Parkinson’s disease, with 755 characteristics and 756 samples from 188 people overall. Table 2 describes the dataset in detail. The data for this study came from 188 PD patients (107 men and 81 women) with ages ranging from 33 to 87 at the Department of Neurology at the Cerrahpaşa College of Medicine, Istanbul University. The control group comprises 64 healthy individuals, ranging in age from 41 to 82, with 23 men and 41 women making up the group. Each participant’s sustained phonation of the vowel /a/ was recorded three times with the microphone adjusted to 44.1 kHz after the doctor’s examination in order to collect data. A variety of speech signal processing algorithms, such as Wavelet Transform-Based Features, Time–Frequency Features, Mel-Frequency Cepstral Coefficients (MFCCs), Vocal Fold Features, and TWQT Features, have been applied to speech recordings of Parkinson’s disease (PD) patients in order to extract clinically relevant information for PD assessment. The dataset contains a total of 755 columns, each representing different features related to voice signals. The baseline features are described in columns 3–23, while intensity parameters are found in columns 24–26. Columns 27–30 contain formant frequencies, and bandwidth parameters are represented in columns 31–34. The vocal fold characteristics can be found in columns 35–56. Mel-Frequency Cepstral Coefficients’ (MFCCs) features are present in columns 57–140, followed by wavelet features in columns 141–322. Columns 323–754 represent the Three-Quarter Time Wavelet Transform (TQWT) features, and the target class labels are located in column 755. Each of these columns provides valuable insights for the analysis and classification of voice signals to aid in differentiating between healthy and affected individuals in the context of Parkinson’s disease detection.

**Table 2** Experimental analysis

Machine learning techniques	Precision		Recall		F1_Score		Support		Accuracy (%)
	Non-PD	PD	Non-PD	PD	Non-PD	PD	Non-PD	PD	
SVM	0.67	0.81	0.06	0.99	0.12	0.89	31	121	80.92
KNN	0.67	0.81	0.06	0.99	0.12	0.89	31	121	80.26
Logistic regression	0.54	0.84	0.39	0.91	0.49	0.87	31	121	79.47
<b>Random forest classifier</b>	<b>0.83</b>	<b>0.91</b>	<b>0.61</b>	<b>0.97</b>	<b>0.70</b>	<b>0.94</b>	<b>31</b>	<b>121</b>	<b>89.47</b>

### 3.2 Proposed Approach

The techniques used in this research paper are SVM, KNN, Random Forest, and Logistic Regression.

**Support Vector Machine (SVM):** Popular machine learning methods for regression and classification include SVM. It works by locating the optimum hyperplane that splits a dataset into several classes. Because the hyperplane is chosen to minimise the distance between it and the closest data points for each class, support vectors earn their name. The equation for the hyperplane is expressed as

$$w * x + b = 0,$$

where  $x$  represents the input vector,  $b$  represents the bias, and  $w$  represents the weight vector. SVM tries to increase the margin and decrease the chance of misidentification of data points. SVM is significant because it can handle high-dimensional datasets and performs well with both linear and nonlinearly separable data.

**K-Nearest Neighbour (KNN):** It finds the K-nearest neighbours to the given data point before classifying it according to the majority class of those neighbours. The gap between the data points is calculated using a variety of distance metrics, including the Manhattan, Euclidean, and cosine distances. Since KNN is a non-parametric method and does not assume anything about the distribution of the underlying data, it is significant. It is an easy method to understand and apply, and it can handle both continuous and categorical data.

**Random Forest Classifier:** It functions by building a number of decision trees, each of which is trained using various traits and data samples. The output of the Random Forest is then created by combining the predictions given by each decision tree. Random Forest is significant because it can handle high-dimensional datasets with many characteristics, missing data, and outliers. It is also less likely to overfit in comparison to individual decision trees.

**Logistic Regression:** A typical statistical method for binary classification tasks is logistic regression. It works by modelling the probability of a binary answer variable, such as 0 or 1, as a function of one or more predictor variables. The logistic regression model determines the probability that the response variable will have a value of 1 given the predictor variables. The logistic regression formula is shown as follows:

$$\log \text{it}(p) = \log\left(\frac{p}{(1-p)}\right) = b_0 + b_1x_1 + \dots + b_nx_n,$$

where  $b_0$ ,  $b_1\dots$ , and  $b_n$  are the logistic regression model's coefficients,  $x_1\dots$ ,  $x_n$  are the predictor variables, and  $p$  is the likelihood that the response variable will have the value 1. Logistic regression is significant because it is a simple yet powerful method that can handle both continuous and categorical predictor variables. Logistic

regression assumes a linear relationship between the predictor elements and the log odds of the response variable, albeit it may not always be appropriate. Furthermore, it makes the assumption that each observation has a uniform distribution and is independent, which may not be true for all datasets, observations, and conclusions.

## 4 Results and Discussions

### 4.1 Performance Matrices

A classification report can be used to evaluate the effectiveness of a classification model. It provides a summary of numerous evaluation metrics for each class, including precision, recall, F1-score, and support. The report usually appears as a table and includes the information below for each class:

- Precision: true positive predictions as a percentage of all positive forecasts.

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

- Recall: the percentage of accurate predictions across all cases where predictions came true.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}).$$

- F1-score: the harmonic average of recall and accuracy.

$$\text{F1-score} = 2 * (\text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}).$$

- Support: the test set's actual number of occurrences of the class
- Accuracy: compares the proportion of correctly classified data instances to all data instances.

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

One model's performance can be assessed using the categorization report over a variety of classes or to compare the efficacy of several classification models. It is also possible to identify the classes that are more challenging to anticipate effectively, and this knowledge can help guide future model advancements.

## 4.2 Experimental Analysis

SVM is the first model that we have used. The hyperparameters of the model have a significant impact on how well it performs. These are the parameters that the SVM model is affected by.

- C: manages the trade-off between minimising the margin and obtaining a low training error. We have kept the C value at 50.
- The type of kernel function used to alter the input data is specified by the kernel property. Radial basis function (RBF) kernels, polynomial kernels, and linear kernels are popular options. We have used RBF for this study.
- Gamma: analyses how each training example affects the decision boundary. We have ‘scale’ as the parameter. It indicates that the gamma value will be determined using the input data’s scale.

The second ML model used in this study is KNN. The list of parameters that affect the KNN model is given below.

- N\_neighbours: the number of neighbours taken into account when classifying is specified by this option. We have set the number to 100.
- Weights: it explains how the contributions of the neighbours are weighed when making a prediction. We have used ‘uniform’ as the parameter; this means that each neighbour’s vote has the same influence on the final classification decision.
- Metric: we have used ‘Manhattan’ as the distance metric.

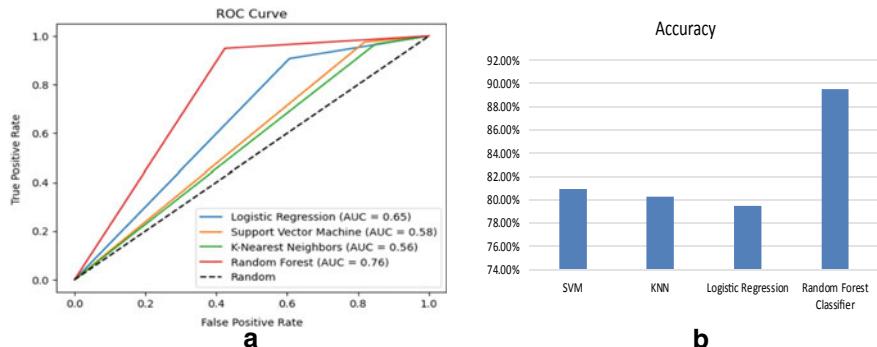
The next model in the list is Logistic Regression. The list of parameters we have used is as follows:

- C: the inverse of regularisation strength. We have set it to 10 for better regularisation.
- Determines the kind of regularisation to use as a penalty. ‘l1’ and ‘l2’ stand for L1 and L2 regularisations, respectively. We have used l2 regularisations.
- Solver: the appropriate optimisation algorithm. We have passed ‘saga’ as the parameter which is efficient for large datasets.

The last ML model used is Random Forest Classifier. Parameters affecting the Random Forest model are as follows:

- n\_estimators: indicates how many trees there are in the forest. Higher values can enhance performance but come at the expense of more complex processing. We have set n as 200.
- max\_features: defines the most features that should be considered while finding the best split. We have supplied the argument as ‘sqrt’. When max\_features is set to ‘sqrt’, the square root of the total number of features in the dataset will be considered when splitting at each node.

We have generated the Receiver Operating Characteristic (ROC) curve to compare the accuracy attained from the aforementioned models. The performance trade-off



**Fig. 3** **a** AUC-ROC to compare different ML techniques. **b** Comparing accuracies from different models

between a binary classification model's true positive rate and the false positive rate is shown by the ROC curve. The area under the curve (AUC-ROC) measures how well the model can distinguish, with a larger number indicating better performance. A bar chart shown in Fig. 3 showcases the accuracies achieved by each individual model.

## 5 Conclusion

Parkinson's disease (PD) is a common chronic neurodegenerative disease characterised by progressive nervous system degeneration. This condition is more prevalent in the elderly. In Parkinson's disease, dopaminergic neurons die early, resulting in a progressive neurodegenerative condition. Parkinson's disease (PD) can cause a wide range of non-motor and motor symptoms, including speech and smell. One of the difficulties that Parkinson's patients may face is a change in speech or difficulty speaking. This study evaluates an automated end-to-end classification strategy for early Parkinson's disease detection. This method uses a classification dataset of voice features to diagnose Parkinson's disease. KBest is applied to the dataset for the feature selection. It selects the top 400 features out of 755. Four different ML models are applied to the dataset naming SVM, KNN, Logistic Regression, and Random Forest Classifier. The accuracies achieved from these models are as follows: 80.92, 80.26, 79.47, and 89.47%, where Random Forest gives the best accuracy among them. The next course of action will be to implement a neural network model and combining them with the existing ML model to create a new hybrid classifier and improve the accuracy that is being achieved. The final goal of this research is to train the model and get higher accuracy than the past already existing model.

## References

1. Tsanas A, Little MA, McSharry PE, Ramig LO (2010) Accurate telemonitoring of parkinsons disease progression by noninvasive speech tests. *IEEE Trans Biomed Eng* 57(4):884–893. <https://doi.org/10.1109/TBME.2009.2036000>
2. Quan C, Ren K, Luo Z, Chen Z, Ling Y (2022) End-to-end deep learning approach for Parkinson's disease detection from speech signals. *Biocybern Biomed Eng* 42(2):556–574. <https://doi.org/10.1016/j.bbe.2022.04.002>
3. Jyotiyan M, Kesswani N, Kumar M (2022) A deep learning approach for classification and diagnosis of Parkinson's disease. *Soft Comput* 26(18):9155–9165. <https://doi.org/10.1007/s00500-022-07275-6>
4. Roobini MS, Reddy YRK, Royal USG, Singh AK, Babu K (2022) Parkinson's disease detection using machine learning. In: 2022 International conference on communication, computing and internet of things, IC3IoT 2022—Proceedings. IEEE. <https://doi.org/10.1109/IC3IOT53935.2022.9768002>
5. Grover S, Bhartia S, Akshama, Yadav A, Seeja KR (2018) Predicting severity of Parkinson's disease using deep learning. *Procedia Comput Sci* 1788–1794. <https://doi.org/10.1016/j.procs.2018.05.154>
6. Tai YC, Bryan PG, Loayza F, Peláez E (2021) A voice analysis approach for recognizing Parkinson's disease patterns. *IFAC-PapersOnLine* 382–387. <https://doi.org/10.1016/j.ifacol.2021.10.286>
7. Khaskhoussy R, Ben Ayed Y (2022) Speech processing for early Parkinson's disease diagnosis: machine learning and deep learning-based approach. *Soc Netw Anal Min* 12(1). <https://doi.org/10.1007/s13278-022-00905-9>
8. Zhang R, Jia J, Zhang R (2022) EEG analysis of Parkinson's disease using time–frequency analysis and deep learning. *Biomed Signal Process Control* 78. <https://doi.org/10.1016/j.bspc.2022.103883>
9. Er MB, Isik E, Isik I (2021) Parkinson's detection based on combined CNN and LSTM using enhanced speech signals with variational mode decomposition. *Biomed Signal Process Control* 70. <https://doi.org/10.1016/j.bspc.2021.103006>
10. Uppalapati B, Srinivasa Rao S, Srinivasa Rao P (2023) Application of ANN combined with machine learning for early recognition of Parkinson's disease. In: Lecture notes in networks and systems. Springer Science and Business Media Deutschland GmbH, pp 39–49. [https://doi.org/10.1007/978-981-19-4863-3\\_4](https://doi.org/10.1007/978-981-19-4863-3_4)
11. Nilashi M et al (2023) Early diagnosis of Parkinson's disease: a combined method using deep learning and neuro-fuzzy techniques. *Comput Biol Chem* 102. <https://doi.org/10.1016/j.compbiochem.2022.107788>
12. Rahman A, Khan A, Raza AA (2020) Parkinson's disease detection based on signal processing algorithms and machine learning. CRPASE: Trans Electr Electron Comput Eng 6(3):141–145 [Online]. Available: <http://www.crpase.com>
13. Quan C, Ren K, Luo Z (2021) A deep learning based method for Parkinson's disease detection using dynamic features of speech. *IEEE Access* 9:10239–10252. <https://doi.org/10.1109/ACCESS.2021.3051432>
14. Iyengar SS (2019) 2019 Twelfth International conference on contemporary computing (IC3–2019) 8–10 August 2019. Jaypee Institute of Information Technology, Noida, India
15. Goyal J, Khandnor P, Aseri TC (2021) A hybrid approach for Parkinson's disease diagnosis with resonance and time-frequency based features from speech signals. *Expert Syst Appl* 182. <https://doi.org/10.1016/j.eswa.2021.115283>
16. Ma J et al (2021) Deep dual-side learning ensemble model for Parkinson speech recognition. *Biomed Signal Process Control* 69. <https://doi.org/10.1016/j.bspc.2021.102849>
17. Chang LC et al. Machine learning approaches to identify Parkinson's disease using voice signal features

# Performance Analysis of Deep CNN, YOLO, and LeNet for Handwritten Digit Classification



Jibok Sarmah, Madan Lal Saini, Ankush Kumar, and Vidhan Chasta

**Abstract** The fundamental applications of handwritten digit classification are in the fields of optical reorganization of digits, bank check processing, recognizing zip codes on mail for postal, processing bank check amounts, and numeric entries in forms filled up by hand. For processing these types of tasks, different kinds of learning algorithms are used. A comparative study on performance analysis was done for convolutional neural network, LeNet-5, and YOLOv7. Publicly available MNIST, DIDA, and MNIST MIX handwritten digit dataset were used in experimental work. The objective of this study is to find the best algorithm which can give an acceptable accuracy. To implement the model, this paper uses a deep neural network CNN and its architecture LeNet-5 and YOLOv7 which have become a potent tool for image categorization problems in recent years. This paper demonstrates the efficacy of deep learning approaches for effective and precise digit recognition, which can be expanded to numerous real-world applications needing accurate and dependable recognition of digits written on paper. This research has achieved the highest accuracy of 99.38% for LeNet-5.

**Keywords** Handwritten digit recognition · Convolutional neural networks · LeNet-5 · YOLOv7 · MNIST

---

J. Sarmah () · M. L. Saini · A. Kumar · V. Chasta

Department of CSE Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India  
e-mail: [jiboksarmah10@gmail.com](mailto:jiboksarmah10@gmail.com)

M. L. Saini  
e-mail: [mlsaini@gmail.com](mailto:mlsaini@gmail.com)

V. Chasta  
e-mail: [21bcs6455@cuchd.in](mailto:21bcs6455@cuchd.in)

## 1 Introduction

The accurate classification of digits written by hand into the corresponding number labels presents a significant difficulty in computer vision. The objective is to develop a trustworthy and efficient model that can recognize and distinguish between numbers from zero to nine. In order to get higher accuracy in digit classification, this problem statement aims to study how convolutional neural networks (CNNs) can be utilized on the data present in MNIST, DIDA, and MNIST MIX dataset. Making a functioning CNN architecture, tuning its hyper parameters, and putting the proper controls in place to decrease overfitting and promote generalization are challenging tasks [1]. To enable applications like analysis of document, optical character recognition, and automatic mail sorting, the goal is construction of a model that can accurately classify handwritten digits in real-world scenarios.

Various CNN models were investigated and assessed in order to find the best model for digit recognition but mostly on single dataset. This involves three datasets and experiments with various arrangements of fully connected, convolutional and pooling layers, as well as with various regularization strategies and activation functions. Secondly, the CNN model's hyper parameters should be optimized. To improve the model's training performance and effectiveness, hyper parameters like rate of learning, batch size, and selection of optimizer will be carefully calibrated. Grid search and random search are two techniques that may be used to carefully examine the hyper parameter space. Any possible overfitting problems will be reduced by using appropriate regularization techniques. Regularization methods like dropout, L1 regularization, L2 regularization, or batch normalization will be looked at in order to increase the model's ability to generalize and prevent it from remembering the training data. As part of the study's objective, the performance of the developed CNN model is also being assessed. This will be accomplished by training the model using the MNIST training dataset and testing it using the MNIST test dataset to assess its performance in terms of accuracy, recall, precision, and other important performance metrics. The model's effectiveness will be evaluated against that of other cutting-edge approaches in order to assess its efficacy and competitiveness. Thus, the study goal finally aims to improve digit classification methods by developing an optimized CNN model for accurate and efficient recognition of handwritten digits. The results of this work might be used in analysis of documents, automatic mail sorting, optical character recognition, and many other sectors that need precise digit recognition.

Due to differences in writing styles, levels of clarity, and possible noise or distortions in the data, it is difficult to recognize handwritten figures. It is difficult to create strong algorithms that can correctly categorize digits in a variety of situations since every individual has a very different style of handwriting. As deep learning and machine learning have advanced, researchers have created a number of methods and algorithms to address the problem of digit classification. The MNIST dataset is a well-known example of a significant dataset that has significantly advanced research in this field. The MNIST dataset is a massive collection of grayscale photographs,

each of which depicts a handwritten digit from 0 to 9, together with its matching label. It is now a common benchmark dataset for assessing how well digit classification algorithms perform.

## 2 Literature Review

Several methods have been investigated for classifying digits, including well-known machine learning algorithms like random forests [2], and support vector machines (SVMs) [3]. Although these techniques have a respectable level of accuracy, the development of deep learning, particularly CNNs, has completely changed the area. The algorithms involved in this review are KNN, SVM, random forest, YOLO, LeNet, and CNN. Authors proposed a deep large simple neural network model to detect the digit and achieved high accuracy on the available dataset. They have used a combination of pooling and convolutional layers to extract features from the given image, and then use fully connected layers for classification.

**LeNet-5 Architecture:** The LeNet-5 architecture by Yann LeCun is among the most significant researches on CNN-based digit classification [4]. LeNet-5 laid the foundation for subsequent improvements by showcasing CNNs' effectiveness in recognizing handwritten digits. There are several, pooling, convolutional, and fully connected layers in it. Study on neural network is proposed by Simard et al. [5], Handwritten Digit Recognition Using CNN. They tried different architectures and achieved over 99% accuracy on the dataset. Study on classifiers is proposed by Sakshica and Gupta [6] and Al Mansoori and Al Mansoori [7], a digit recognition method based on ANN and CNN. They extract features from images using a technique named histogram of oriented gradients (HOG) and achieve > 98% accuracy. Handwritten digit recognition using a MNIST dataset [8], they extracted features using a gray level co-occurrence matrix (GLCM) and fulfilled over 98% accuracy on the dataset.

Digit recognition method based on CNN is proposed by Hossain and Ali [9] and in their work, they used YOLO for object detection and CNN for classification. They have fulfilled over 99% accuracy on the dataset. Authors [10] used a parametric technique for removing internal covariate shift in deep networks. Authors [11] used multi-column deep neural networks for image classification and authors [13] used hierarchical softmax in their CNN models. Authors [12] presented a comparison on performance of various hidden layers of CNN. Authors [14] used image segmentation using expectation maximization in their CNN model. A real-time handwritten digit recognition was done by Ahamed et al. [15] using SVM.

The literature survey shows that ML algorithms and DL algorithms are vibrantly used for handwritten digit recognition. Among the algorithms discussed in this study, convolutional neural networks (CNNs) showed the highest accuracy, achieving over 99% accuracy on the MNIST dataset. However, other algorithms such as KNN, SVM, random forest, and YOLO have also shown promising results and can be used for digit recognition in certain scenarios.

### 3 Objective, Motivation and Challenges

The capacity to correctly identify and categorize handwritten numbers is extremely important in today's digital world across many different fields. Precise digit categorization is essential for increasing automation and efficiency, from digit-based document analysis to automated mail sorting. The process of detecting handwritten digits and giving them the appropriate numerical labels is known as "digit classification." Due to its numerous practical applications, it is a basic issue related to the field of machine vision and has attracted a lot of interest. For jobs like optical character recognition (OCR), where it's important to extract useful information from handwritten papers or forms, accurate digit categorization is essential. Additionally, to efficiently process high numbers of mail items, automatic mail sorting systems significantly rely on digit recognition. The main objective of this paper is to find the suitable algorithm for handwritten digit recognition and to compare the accuracy of different models along with their execution time to get the best possible model.

We applied some algorithms to read some handwritten documents but we did not get satisfactory results, so we were having a desire to perform an analysis on all the algorithms for handwritten digit recognition. This desire motivated us to perform this research. We face the main challenges of dataset, only few datasets are publically available and mostly on digits, not on English letters.

### 4 Methodology

To get the greatest outcomes, we utilized a variety of tools and strategies throughout the project. For the development of the CNN approach with five layers and the MNIST, DIDA and MNIST MIX dataset were used. Keras, Tensorflow, and Python were used for model building. The CNN and its popular architecture LeNet were used. It is far more effective than other neural network techniques for locating and identifying items in pictures. How this operates is clearly explained in this article.

#### 4.1 *Dataset Description*

##### MNIST Dataset

The term "Modified National Institute of Standards & Technology" is also written as MNIST. It is a huge dataset that is commonly used in convolutional neural networks models and computer vision. Systems that can use numerical data as input are tested and trained using the MNIST dataset. Images of manually written digits are stored on this. 60,000 photos make up this dataset, which is used for cross validation and training. Additionally, 10,000 additional photos might be examined. The dataset has a total of 784 pictures, of dimension vector of  $28 * 28$ . Figure 1 shows how each



**Fig. 1** MNIST dataset

number's size of  $28 * 28$  creates a reliable and recognizable combination that makes it simple to identify the handwritten numbers [8].

### DIDA and MNIST MIX Dataset

DIDA dataset is a specialized dataset and this is especially chosen for project or research study. The dataset consists of a number of photos in grayscale that represents handwritten digits from 0 to 9 and every digit is represented by  $28 \times 28$  pixels. MNIST MIX dataset is a multi-language handwritten digit recognition dataset. It is having mix digits from many languages and makes it the largest dataset which has the same type with reference to both data samples and languages.

## 5 Implementation

### 5.1 Convolution Neural Network

A popular deep learning neural network structure in computer vision is convolutional neural networks. A pooling layer, a fully connected (FC) layer, and convolutional layer are the three basic layers that make up CNN. The first layer is convolutional, while the last layer is FC. The complexity of the layers increases for the first layer to the last layer in CNN model. Because of the rising complexity, CNN (network model) can understand increasingly complex parts of a picture until it is able to recognize the complete thing.

## 5.2 Convolutional Layer

The fundamental element of the convolutional neural network (CNN), which is in charge of carrying out the majority of calculations, is the convolutional layer. It is possible to stack many convolutional layers on top of one another to improve the network's capacity to extract intricate information from the input pictures. A kernel or filter within the convolutional component travels over the receptive fields of the input picture during the convolution process. This kernel moves across multiple places and orientations as it systematically analyzes the whole image. Its goal is to find particular characteristics or patterns that the system is programmed to recognize. A dot product operation is carried out between the filter and the appropriate input pixels inside its receptive field at each iteration. The degree of resemblance or correlation between the filtered area and the local picture patch is determined by this dot product operation. The convolutional layer creates feature maps or a convolved feature by traversing the whole picture in this way. The feature map displays the spatial pattern of feature detection over the whole picture. It is created by joining the dots together or carefully mixing the computed dot products. This procedure aids in highlighting and suppressing portions of the image that have the required attributes. The picture is converted to numerical values under the convolutional layer so that the CNN may examine and comprehend it. The network may execute mathematical operations and look for significant patterns or correlations among the input pixels by describing the picture as a numerical array. The CNN can efficiently learn from the input data and extract higher-level representations, thanks to this conversion of the picture into numerical values. After learning patterns and correlations, the network may utilize these representations to generate predictions or categorize data.

## 5.3 Pooling Layer

As in the first layer, this layer is essential to convolutional neural networks (CNNs) for processing images. It works by downsampling the input picture before applying a kernel or filter, usually in the form of a filter. With the use of this layer, the feature maps' spatial dimensions may be shrunk while still retaining key details. It's crucial to remember that the pooling layer also causes an information loss. The input areas are downsampled or made smaller during the pooling process, which causes this loss. As a result, this downsampling procedure might result in the loss of certain fine-grained features. Despite the information loss, the pooling layer helps to the CNN's general efficacy and efficiency. The pooling layer contributes to streamlining the network design and improving its computing efficiency by lowering the spatial dimensions. Furthermore, pooling improves the network's capacity to identify and isolate the most crucial and conspicuous elements from the input pictures. Applying pooling techniques like max pooling or average pooling, which summarize and aggregate the data within a small neighborhood, allows for this decrease in parameters.

## 5.4 Fully Connected Layer

In a CNN, the fully connected (FC) layer classifies pictures by using the characteristics that were derived from the layers above. Fully connected refers to a state in which every input or node from a layer before it is connected to every activation unit or node in a layer after it. However, a considerably denser network would result from using completely linked layers over the whole CNN. This density would result in greater computing costs, more losses, and perhaps even lower output quality from the network. In order to balance performance and efficiency, CNNs often use selective connections instead of having all layers fully coupled.

## 5.5 Creating a CNN Model

**Dataset preparation:** First, we took the MNIST dataset—a collection of handwritten digit images and the labels that go with them—and divided it into training and testing sets using an 80/20 split. The images were then preprocessed, scaled to a common size ( $28 \times 28$  pixels), and the pixel values were normalized to range from 0 to 1.

**Model Architecture Design:** The architecture of the CNN model was built as shown in Fig. 2. We first constructed a stack of convolutional layers, followed by activation methods to identify characteristics in the supplied images. After each layer, the spatial dimensions were downsampled and the most crucial data was gathered using a pooling layer. We increased the number of layers in order to make the network deeper, we almost doubled the number of the convolutional layers.

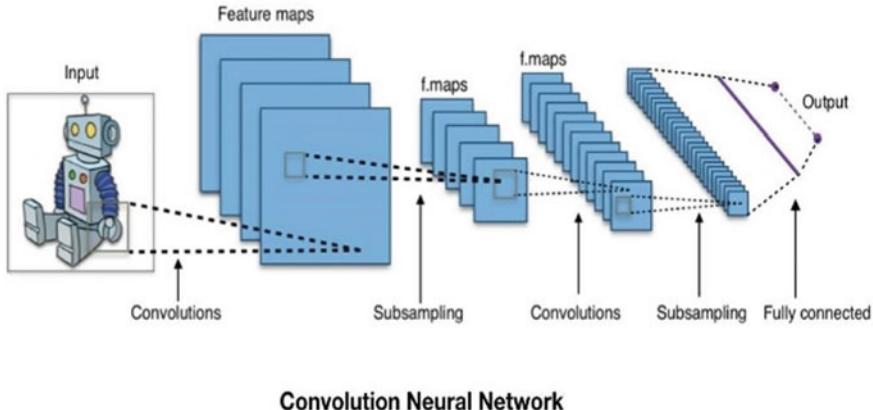
**Flatten and Fully Connected Layers:** Flatten the feature maps into a 1D vector after the convolutional and pooling layers. The completely connected layers of a traditional neural network should be connected to this vector. Apply activation functions to these layers to introduce nonlinearity. The number of digit classes should be the same as the neurons present in the final fully connected layer. (10 neurons for digits 0–9).

**Output Layer and Loss Function:** Each digit class's newly created class probabilities have an output layer with softmax activation. The categorical cross-entropy loss function, which calculates the difference between the actual labels and the anticipated probability, is used for training the designed model after the softmax function ensures that the arithmetic sum of the predicted probability is 1.

**Feedforward Neural Networks:** For digit classification on the MNIST dataset, researchers have extensively employed feedforward neural networks, especially multilayer perceptron (MLPs). To improve classification accuracy, several network designs, activation functions, and optimization methods have been researched. To avoid overfitting, methods like batch normalization and dropout regularization have also been used.

**Model Compilation:** By specifying the learning rate and an optimizer, the model was built. Secondly, we keep an eye on the model's progress as it is being trained to discover the accuracy.

# CNN



**Convolution Neural Network**

**Fig. 2** CNN model

**Model training** applied the model to the training dataset that was supplied. Back-propagation is used to update the model's weights and biases by repeatedly iterating through the training data for a specific number of epochs (in our study, we chose 10 epochs). In order to detect overfitting during training, we monitored performance of the model on a validation set and made the required hyper parameter modifications.

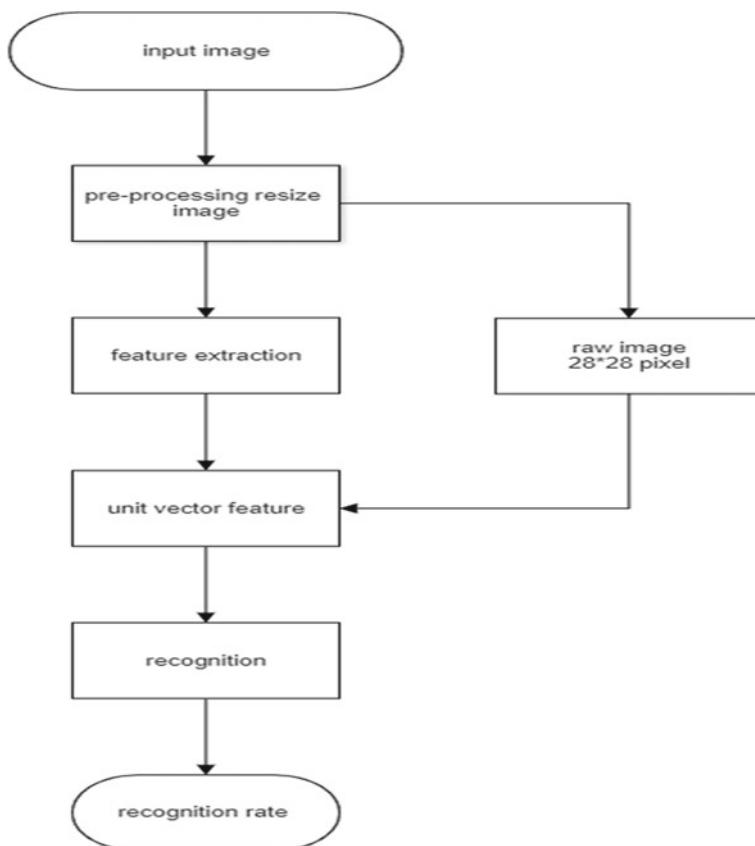
An image segmenter is used to segment and extract the pertinent parts of the picture if it is a hand drawing. If, however, the picture comes from the MNIST dataset, it goes straight to the next stage of processing. The image is created, then it is scanned to extract significant characteristics and patterns that will help with digit classification. The software then incorporates a model that has been retrained after being previously trained on a sizable collection of labeled digit pictures to increase accuracy. More labeled digit pictures can be used to train the model if necessary. Alternately, the programmed switches to the convolutional neural network (CNN) system if more training is not required. The picture is sent into the CNN system's linked layers of artificial neurons, which examine and process it to provide an output probability that represents the expected digit label. The program then processes a CSV file that could contain more data after saving a report that summarizes the categorization procedure. The CSV file processing is finished, and then the program ends its operation, bringing the digit categorization workflow to a close.

**Model Evaluation:** Using the testing dataset, the generalization performance of the training model was examined. Recall, accuracy,  $F_1$  score, and precision were examined metrics to evaluate the model's classification performance. We investigated

the potential issues (under fitting or overfitting), but fortunately we encountered no issues during the model evaluation. Instead, we optimized the model's performance by fine-tuning it by testing different hyper parameters, learning rates, batch sizes, and regularization techniques (like dropout) to improve convergence and keep the model's complexity as low as possible.

Flowchart: The program shown in Fig. 3 uses a sequential approach to classify the digits. Starting with identifying the kind of input picture, it distinguishes between images created manually and those taken from the MNIST collection.

Hyper parameter Optimization: To enhance the functionality of CNN models, researchers have concentrated on hyper parameter optimization. We worked on finding the ideal settings for hyper parameters like learning rate, batch size, and optimizer selection that has been done using approaches such as Bayesian optimization, random, and grid search. Better convergence, less overfitting, and improved generalization are all benefits of hyper parameter optimization.



**Fig. 3** Flowchart for digit reorganization system

**Regularization Methods:** Regularization methods have undergone substantial research in order to reduce overfitting. The generalization capability of CNN models has been improved with dropout, a regularization method that eliminates units at random while training. To increase model resilience, many other methods, such as batch normalization, L1 regularization or L2 regularization, and augmentation of the data have also been investigated.

### **5.6 You Only Look Once (YOLO)**

Preprocess the MNIST, DIDA, and MNIST dataset by converting image to YOLO format. The YOLO version 7 was used for our experimental work. This involves dividing the images to grid and predicting bounding boxes and class probabilities for grid cells. Train the model on basis of training set, this involves optimizing the parameters of neural network and loss function is used to penalize incorrect prediction. Predict the given image of test set using training set, this detects the digit of each image using predicted bounding box and assign the class label which has highest probability.

### **5.7 LeNet**

Large fully connected multilayer neural network has two layers of weights and get trained with various numbers of hidden layers. LeNet version 5 was used for experimental work which has more feature maps than LeNet 4 and it uses distributed representation to encode the categories at the output layer. It has 340 k connections and 60 k parameters.

**Evaluation of Performance and Comparison:** Numerous researches have assessed and contrasted the effectiveness of CNN models on the MNIST. The classification skills of the models have been evaluated using metrics including F1 score, recall, precision, and accuracy. In order to establish benchmarks, comparisons between the CNN-based approaches and other deep learning architectures and traditional methods for machine learning have been made.

## **6 Result and Discussion**

The trained CNN model used the MNIST dataset to classify digits with an accuracy of 97.55% on the testing dataset. This performance shows how well the CNN architecture recognizes and categorizes handwritten digits with accuracy. The YOLOv7 and LeNet-5 models, which attained accuracies of 97.83% and 99.38% respectively, were surpassed by the LeNet-5 model. This demonstrates how effective CNNs are at

collecting intricate spatial cues and getting better classification results on the MNIST dataset. By correctly identifying digits from the testing dataset, the CNN model showed strong generalization skills, demonstrating its capacity to handle untried cases. Additionally, the model demonstrated resilience against differences in various orientations, writing styles, and modest digit image distortions.

The CNN architecture LeNet-5 and YOLOv7 are giving better results when compared with the simple CNN. The LeNet-5 is a better choice than YOLOv7 because it is designed specifically for digit recognition and gives higher accuracy. Accuracies for MNIST MIX dataset are low compared with MNIST and DIDA dataset. This dataset is having mix digits from 10 different languages and this leads to high variability and appearance in writing styles which is a reason of low accuracy.

The training dataset was expanded using data augmentation methods including rotation, scaling, and translation. By decreasing overfitting and enhancing the model's capacity to accommodate differences in digit pictures, this augmentation method helped the model perform better.

The choice of hyper parameters had a significant impact on how well the CNN model performed. The tuning of hyper parameters such as batch size, learning rate, and the number of filters with in the convolutional layers was done in a methodical manner using grid search or random search. The model's accuracy and convergence were considerably aided by the optimized hyper parameters. To understand what the CNN model focused on while categorizing digits, the learnt features inside the convolutional layers of the network were visualized. It was found that the shallow convolutional layers learnt more abstract and discriminative characteristics specific to each digit class, whereas the deeper layers learned low-level features like edges and corners. To comprehend the difficulties the model encountered, a few cases that were incorrectly categorized were examined. It was discovered that cases with unclear or badly written digits, distorted pictures, or digits written in atypical styles that considerably differed from the training examples were common sources of misclassifications. In order to handle such scenarios more effectively, these insights can direct future changes to the model given in Table 1.

Despite obtaining excellent accuracy, the CNN model may still run into problems when writing digits in small font sizes, severely overlapping them, or with significant distortion. To overcome these drawbacks and enhance the model's performance, further study might concentrate on creating innovative structures, including attention processes, or investigating multi-task learning. The MNIST dataset's effective implementation of a CNN model for digit classification has ramifications for a number of

**Table 1** Performance of different models

Algorithms	Accuracy		
	MNIST	Dida	MNIST mix
CNN	97.55%	98.6%	93.7%
YOLOv7	97.83%	98.8%	92.48%
LeNet-5	99.38%	98.73%	93.35%

practical uses. It can be used in bank check digit recognition, digit-based CAPTCHA solution, and automatic mail sorting systems. The model is an effective tool for digit recognition jobs because of its efficiency and accuracy.

## 7 Conclusions

LeNet-5 has demonstrated to be quite effective in digit classification problems. They may automatically learn intricate characteristics from the input photos, which can increase the classification task's accuracy. LeNet-5 has been proven to greatly outperform more conventional machine learning algorithms, such as CNN and YOLOv7, on the MNIST test set. The capacity of LeNet-5 to tolerate alterations in the input pictures, such as rotation, scaling, and translation is one of its key advantages. Pooling layers, convolutional layers, and fully connected layers were used to do this, enabling the network to learn features at various sizes and degrees of abstraction. CNN has acceptable accuracy but has certain drawbacks, such as overfitting, which happens when the model is overly complicated in comparison with the size of the dataset. To overcome this, methods like regularization and dropout can be applied to enhance the model's generalizability and reduce overfitting. DIDA and MNIST MIX dataset are having lower accuracy when compared with the MNIST dataset due to the high variability and appearance in writing styles. The LeNet-5 is a better choice than YOLOv7 because it is designed specifically for digit recognition and gives higher accuracy of 99.38%.

## Future Scope

The future ahead of handwritten digit recognition based on machine and deep learning algorithms is almost limitless. We can work on a denser or hybrid algorithm than the existing collection of algorithms in the future with more diverse data to solve numerous difficulties. More work can be done against differences in various orientations, writing styles, and modest digit image distortions.

## References

1. Wu M, Zhang Z (2010) Handwritten digit classification using the MNIST data set," no September, pp 1–9 (Online). Available: [https://www.researchgate.net/profile/MingWu23/publication/228685853Handwritten\\_Digit\\_Classification\\_using\\_theMNISTData\\_Set](https://www.researchgate.net/profile/MingWu23/publication/228685853Handwritten_Digit_Classification_using_theMNISTData_Set)
2. Pandey P, Gupta R, Khan M, Iqbal S (2020) Multi-digit number classification using MNIST and ANN. Int J Eng Res V9(05):415–421. <https://doi.org/10.17577/ijerty9is050330>
3. Islam KT, Mujtaba G, Raj RG, Nweke HF (2017) Handwritten digits recognition with artificial neural network. In: 2017 International Conference Engineering Technology and Entrepreneurship, ICE2T 2017, vol. 2017-Janua, no. January 2018, pp. 1–4. <https://doi.org/10.1109/ICE2T.2017.8215993>

4. Farabet C, Poulet C, LeCun Y (2009) An FPGA-based stream processor for embedded real-time vision with convolutional networks. In: 2009 IEEE 12th international conference on computer vision workshops, ICCV Work, pp 878–885. <https://doi.org/10.1109/ICCVW.2009.5457611>
5. Simard PY, Steinkraus D, Platt JC (2003) Best practices for convolutional neural networks applied to visual document analysis. In: Proceedings of international conference on doctor analysis and recognition, ICDAR, vol 2003-Janua, no September, pp 958–963. <https://doi.org/10.1109/ICDAR.2003.1227801>
6. Sakshica D, Gupta DK (2015) Handwritten digit recognition using various neural network approaches. IJARCCE 4(2):78–80. <https://doi.org/10.17148/ijarcce.2015.4218>
7. Al Mansoori S, AL-Mansoori S (2015) Intelligent handwritten digit recognition using artificial neural network. J Eng Res Appl 5(June):46–51. [www.ijera.com](http://www.ijera.com). <https://doi.org/10.13140/RG.2.1.2466.0649>
8. Deng L (2012) The MNIST database of handwritten digit images for machine learning research. IEEE Signal Process Mag 29(6):141–142. <https://doi.org/10.1109/MSP.2012.2211477>
9. Hossain MA, Ali MM (2019) Recognition of handwritten digit using convolutional neural network (CNN). Glob J Comput Sci Technol 19(May):27–33. <https://doi.org/10.34257/gjstd.vol19is2pg27>
10. Arpit D, Zhou Y, Kota BU, Govindaraju V (2016) Normalization propagation: a parametric technique for removing internal covariate shift in deep networks. In: 33rd International Conference on Machine Learning ICML 2016, vol 3, no 2015, pp 1800–1810
11. Ciregan D, Meier U, Schmidhuber J (2012) Multi-column deep neural networks for image classification. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit February:3642–3649. <https://doi.org/10.1109/CVPR.2012.6248110>
12. Siddique F, Sakib S, Siddique MAB (2019) Recognition of handwritten digit using convolutional neural network in python with tensorflow and comparison of performance for various hidden layers. In: 2019 5th International conference on advances in electrical engineering ICAEE 2019, pp 541–546. <https://doi.org/10.1109/ICAEE48663.2019.8975496>
13. Liu Q et al (2022) Hierarchical softmax for end-to-end low-resource multilingual speech recognition, (Online). Available: <http://arxiv.org/abs/2204.03855>
14. Carson C, Belongie S, Greenspan H, Malik J (1999) Blobworld : image segmentation using expectation-maximization and its application to image querying 1 introduction feature extraction, pp 1–16
15. Ahamed H, Alam I, Islam M (2019) SVM based real time hand-written digit recognition system SVM based real time hand-written digit recognition system keywords, (Online). Available: <https://www.researchgate.net/publication/330684489>

# Data-Driven Decision Support Systems in E-Governance: Leveraging AI for Policymaking



Anudeep Arora, Prashant Vats, Neha Tomer, Ranjeeta Kaur, Ashok Kumar Saini, Sayar Singh Shekhawat, and Monika Roopak

**Abstract** Data-driven decision support systems have been used more and more in e-governance as a result of the digital revolution. In order to improve the efficacy and efficiency of policymaking, this research article investigates the integration of artificial intelligence (AI) approaches into e-governance systems. Governments can access enormous volumes of data, and AI algorithms are used to analyze and extract insightful data that enables decision-making based on facts. The article emphasizes the advantages of using AI in the e-governance space while formulating policy. Decision support systems can analyze and understand complicated information by utilizing cutting-edge machine learning and data analytics approaches, revealing trends, patterns, and correlations that would be challenging for human analysts to manually find. As a result, decision-makers in government may make well-informed choices based on impartial research and data. The paper also examines the difficulties and factors to be considered when implementing AI in decision support systems for e-governance. With an emphasis on the significance of responsible AI governance frameworks, ethical issues, including algorithmic bias, transparency, and accountability are addressed. The article also explores the effects of incorporating AI into decision-making processes, including potential sociopolitical effects and the requirement for stakeholder participation and public confidence. The results of this study

---

A. Arora · R. Kaur

Kamal Institute of Higher Education and Advance Technology, GGSIPU, New Delhi, India

P. Vats (✉) · A. K. Saini · S. S. Shekhawat

Department of CSE, SCSE, Manipal University Jaipur, Jaipur, Rajasthan, India

e-mail: [prashantvats12345@gmail.com](mailto:prashantvats12345@gmail.com)

N. Tomer

ICFAI Business School, ICFAI University, Dehradun, Uttarakhand, India

e-mail: [neha@iudehradun.edu.in](mailto:neha@iudehradun.edu.in)

M. Roopak

Artificial Intelligence and Cyber Security, Department of Computer Science, School of Computing and Engineering, University of Huddersfield, Huddersfield, UK

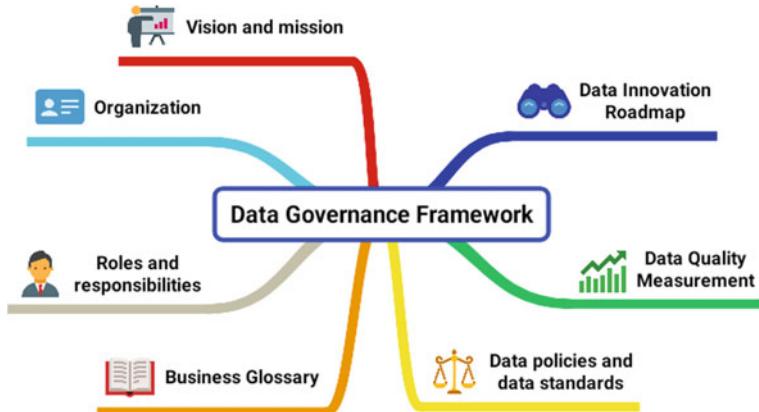
e-mail: [M.Roopak@hud.ac.uk](mailto:M.Roopak@hud.ac.uk)

show how data-driven decision support systems may revolutionize e-governance policies when equipped with AI technology. Governments may enhance their decision-making processes and the outcomes of governance by using the power of big data and sophisticated analytics. This will lead to better public service delivery.

**Keywords** Decision support system · E-governance · Artificial intelligence · Blockchain · Data security · Policymaking

## 1 Introduction

Effective policymaking is essential to providing effective public services and addressing societal concerns in the area of governance, as seen in Fig. 1. Governments have adopted electronic platforms and data-driven systems more and more as a result of the development of digital technology, streamlining administrative procedures and boosting citizen involvement [1, 2]. A potential way to revolutionize policymaking and decision support has developed in recent years with the incorporation of artificial intelligence (AI) technology into e-governance platforms [3–5]. Governments' ability to analyze, understand, and use the huge volumes of data at their disposal might be completely changed by data-driven decision support systems that are powered by AI algorithms. Traditional approaches to policymaking frequently rely on human analysts painstakingly sorting through vast amounts of data, a process that is time-consuming and liable to biases and errors [6–8]. But because of developments in AI and machine learning, decision support systems can now analyze and extract important insights from large, complicated datasets more quickly, precisely, and objectively. This research paper's main goal is to investigate how data-driven decision support systems related to e-governance and how they could revolutionize the way policies are made [9, 10]. Governments may make intelligent, fact-based choices by utilizing AI approaches to take advantage of large data analytics, pattern recognition, and predictive modeling. This essay will examine the advantages and possibilities provided by the incorporation of AI into e-governance policymaking [11, 12]. It will go through how data-driven decision support systems may help decision-makers spot trends as they emerge, assess their options for policy, and foresee the possible effects of different measures [13–15]. Governments may abandon conventional methods and adopt evidence-based policymaking that is based on empirical insights and unbiased analysis, thanks to our capacity to handle and analyze enormous volumes of data. The article will discuss both the advantages of using AI in decision support systems for e-governance as well as the drawbacks and considerations that come with its use [16–18]. To emphasize the significance of responsible AI governance frameworks, ethical issues such as algorithmic bias, transparency, accountability, and privacy will be covered. In addition, the sociopolitical ramifications of using AI in policymaking will be studied, as well as the necessity of stakeholder participation and public confidence [19]. This study intends to shed light on how data-driven decision support systems, backed by AI technology, have the potential to revolutionize policymaking



**Fig. 1** To show the realm of data governance

in e-governance through an examination of the current literature, case studies, and best practices [20, 21]. The research will advance the discussion on how to use AI to innovate in the public sector and improve governance, which will eventually lead to more effective and significant policy development and execution [22].

## 2 Related Work

We will present the work that is connected to data-driven decision support systems in e-government in this area.

In their study [23], Butterworth et al. look at the benefits and drawbacks of using AI in policymaking. It talks about how decision support systems powered by AI could improve policy analysis, stakeholder participation, and decision-making transparency. The study also emphasizes how crucial it is to address ethical issues and build public confidence in AI-driven policies.

An overview of current trends, difficulties, and prospects concerning AI in e-governance is given in Casares et al.'s article [24]. The potential of AI in policy-making and decision support is examined, highlighting the significance of data-driven methodologies. The study highlights effective AI deployments in e-governance through case studies and best practices.

The relationship between artificial intelligence and e-governance is examined by Erka et al. [25]. It gives a summary of the current research on the use of AI in many e-governance areas, including policymaking. The report points up areas for further research and suggests new avenues for using AI in decision support systems for e-governance.

The potential and difficulties of digital transformation are examined in the context of public administration, particularly policymaking, by Marijn et al. [26].

It talks about how artificial intelligence (AI) technologies, such as machine learning and natural language processing have the potential to improve decision support systems and policy results. The study emphasizes the necessity of morally sound AI governance and e-government.

A review of the literature on artificial intelligence in e-government, with a focus on decision-making and policymaking, is given by Kouziokas et al. [27]. It summarizes research results, highlights major topics and knowledge gaps, and suggests a study plan for the future. The study emphasizes the potential of AI-driven decision support systems in enhancing e-governance policymaking procedures.

An overview of how data and digital technologies are changing policymaking processes is given by Chih-Hao et al. [28], who also highlight the potential of data-driven decision support systems to enhance policy results.

The use of ICT applications, such as data-driven decision support systems, in fostering sustainable development through e-governance is explored by Liu et al. [29]. In the context of democratic and environmental governance, it analyzes how artificial intelligence (AI) technology might improve policymaking processes.

The compatibility of data-driven decision support systems with societal values is examined by Matheus et al. [30]. It emphasizes the necessity of open and accountable systems and explores the significance of embedding social values into AI-based policymaking processes.

A bibliometric study of scholarly works on AI in public administration is given by Sun et al. [31]. It provides information on developing issues, approaches, and research trends in the area of e-governance using AI-driven decision support systems.

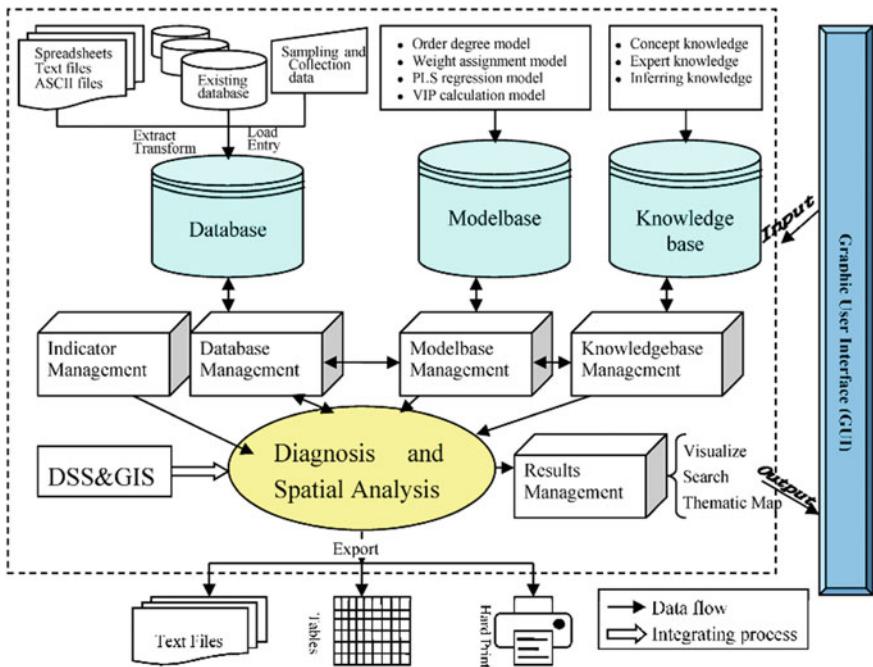
The advantages and difficulties of open data are discussed in the context of e-governance by Wirtz et al. in their article [32]. It emphasizes how open data may be used to improve the delivery of public services and data-driven decision support systems that are assisted by AI.

Brynjolfsson et al.'s [33] analyze of the potential and problems associated with government open data programs. It talks about how AI-enabled data-driven decision support systems may use open data to give useful insights for public administration and policymaking.

These connected studies [34] offer insights into several features of data-driven decision support systems in e-governance, including the role of AI, conformity with public ideals, open data concerns, and the more extensive implications for policy-making processes. They add to the amount of information already available and serve as a starting point for further studies in this area.

### 3 Proposed Work

As illustrated in Fig. 2, the proposed study's objectives are to examine the function of data-driven decision support systems in e-governance and analyze how AI technologies may be successfully used for policymaking procedures [35, 36]. The



**Fig. 2** To show the proposed framework for the data-driven decision support systems in e-governance

study will concentrate on the possible advantages, difficulties, and effects of incorporating artificial intelligence into decision support systems in the context of electronic governance [37].

### 3.1 Research Objectives

1. To investigate the most cutting-edge AI and data-driven decision support systems currently used in e-governance.
2. To list the major obstacles and factors that should be considered when incorporating AI into the process of formulating policy.
3. To investigate the advantages and chances that might come from using AI for e-governance policy analysis and decision assistance.
4. To create a theoretical foundation for incorporating AI technology into data-driven decision support systems for e-governance policymaking.
5. Conduct empirical research, such as case studies and data analysis to assess the efficiency and influence of AI-driven decision support systems in legislative and regulatory procedures.

6. To make recommendations for rules and best practices for the ethical and responsible use of AI in e-governance decision support systems.

### ***3.2 Research Methodology***

The study will combine a literature review, case studies, data analysis, and stakeholder interviews. To identify pertinent ideas, frameworks, and issues relating to data-driven decision support systems and AI in e-governance, a thorough examination of the body of current literature and case studies will be conducted during the first phase [38–40]. This will act as the starting point for creating a conceptual framework. The following stage will concentrate on empirical research, including performing case studies of governmental organizations that have included AI-driven decision support systems in their policymaking procedures [41–44]. Interviews with decision-makers, users of the system, and other interested parties will be used to gather data. In order to assess the influence of AI on the results of policy, quantitative analysis of pertinent datasets will also be carried out [45, 46]. The study will also include ethical issues including algorithmic bias, accountability, transparency, and privacy related to AI in e-governance. To guarantee the proper use of AI technology in decision support systems, ethical principles and best practices will be suggested [47, 48].

### ***3.3 Proposed Contributions***

By illuminating the potential of data-driven decision support systems aided by AI for policymaking in e-governance, this research seeks to add to the body of current knowledge [49]. The results will provide policymakers, government organizations, and researchers with information regarding the advantages, difficulties, and implications of utilizing AI technology in the decision-making process [50]. The suggested standards and best practices will direct the ethical adoption and application of AI in e-governance, guaranteeing openness, equity, and accountability [51–53].

Overall, this research will deepen our understanding of how AI may improve the policymaking procedures in e-governance, allowing governments to make decisions based on solid facts and produce better governance results [54–56].

## **4 Experimental Results**

To determine how effectively AI-driven decision support systems perform in terms of impact, efficacy, and efficiency, experimental findings frequently include comparing them to more traditional techniques. Quantitative measurements, including accuracy,

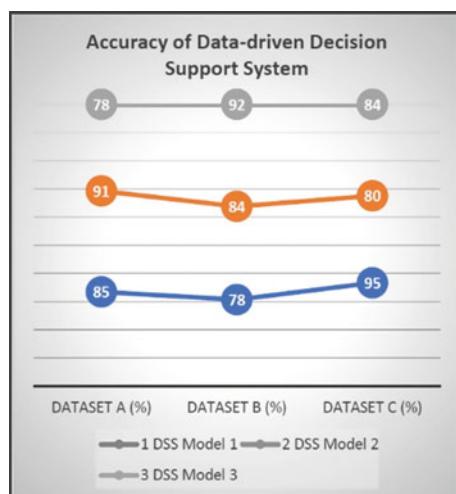
precision, recall, or other performance indicators are included in the results to evaluate the system's capability for data analysis and interpretation, the creation of useful insights, and the support of policy decisions. The findings for data-driven decision support systems in e-governance contribute to the body of knowledge in the field by providing insightful information on the effectiveness of AI-driven decision support systems for policymaking within the context of e-governance. To evaluate the success of data-driven decision support systems (DSS) in e-governance, the following quantitative metrics are utilized, with a focus on their ability to analyze and comprehend data for policymaking.

#### 4.1 Accuracy

This statistic assesses the general accuracy of the predictions or classifications made by the DSS. It shows the proportion of accurately foreseen outcomes or judgments in all possible cases. For instance, the accuracy indicator would be 85% if the DSS correctly predicts 85% of policy outcomes. The outcomes are represented visually in Fig. 3.

Table 1 gives the results of the evaluation of three distinct DSS models using the datasets A, B, and C. The percentage of outcomes that were properly predicted by each DSS model on each dataset is represented by the accuracy values, which are expressed as percentages. Table 1 makes it possible to compare the accuracy performance of the DSS models across various datasets.

**Fig. 3** Accuracy of data-driven decision support system in e-governance



**Table 1** Table for accuracy of data-driven decision support system in e-governance

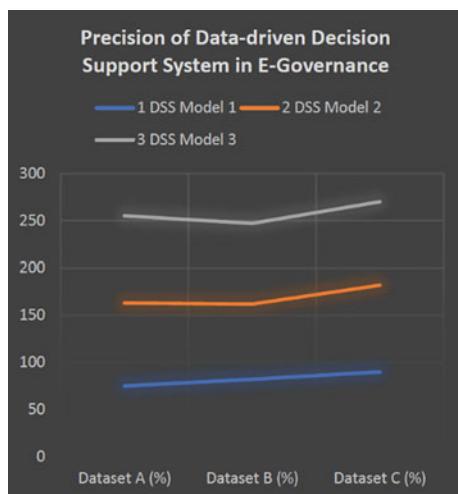
S. No.	System	Dataset A (%)	Dataset B (%)	Dataset C (%)
1	DSS model 1	85	78	90
2	DSS model 2	91	84	87
3	DSS model 3	88	92	89

## 4.2 Precision

Precision represents the percentage of accurate positive predictions among all DSS-generated positive predictions. It shows how effectively the DSS prevents false positives. Having a high precision means that the DSS is producing reliable positive predictions. It may be computed by dividing the total number of true positives by the total number of false positives. In Fig. 4, the findings for the precision of the data-driven decision support system in e-government are displayed.

Table 2 contains the results of the evaluation of three alternative DSS models using datasets A, B, and C. The accuracy numbers, which are shown as percentages, show the share of accurate positive predictions among all positive forecasts generated by each DSS model on each dataset. The table enables a comparison of the accuracy performance across several datasets of the DSS models.

**Fig. 4** Precision of data-driven decision support system in e-governance



**Table 2** Table for precision of data-driven decision support system in e-governance

S. No.	System	Dataset A (%)	Dataset B (%)	Dataset C (%)
1	DSS model 1	75	82	90
2	DSS model 2	88	80	92
3	DSS model 3	92	85	88

**Table 3** To display values for recall (sensitivity) of data-driven decision support system in e-governance

S. No.	System	Dataset A (%)	Dataset B (%)	Dataset C (%)
1	DSS model 1	80	75	88
2	DSS model 2	85	82	90
3	DSS model 3	92	88	94

### 4.3 Recall (Sensitivity)

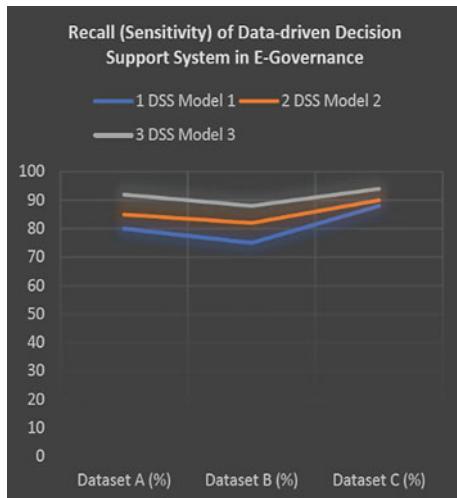
Recall quantifies the percentage of correct positive predictions among all actual positive occurrences. It shows how successfully the DSS prevents false negatives. High recall means that a large portion of positive cases are being captured by the DSS. It may be computed by dividing the total of true positives and false negatives by the number of true positives.

Three alternative DSS models, which were tested using three different datasets (A, B, and C), are presented in Table 3. The percentages of genuine positive predictions out of all real positive cases that each DSS model on each dataset really recorded make up the recall values, which are provided for each dataset. The recall performance of the DSS models across various datasets may be compared using the table. Figures 5 and 6 display the outcomes for recall (sensitivity) of data-driven decision support system in e-government.

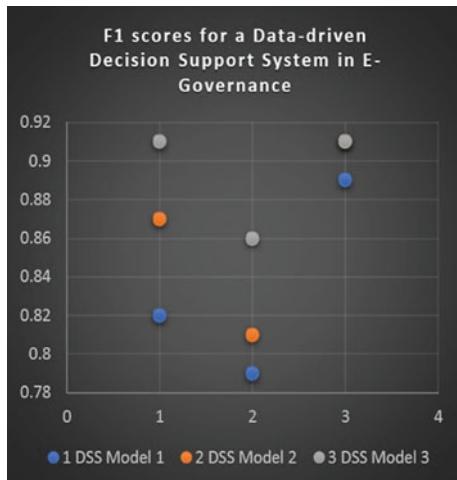
### 4.4 $F_1$ Score

The harmonic mean of recall and accuracy is the  $F_1$  score. It offers a statistic that strikes a compromise between precision and recall. As  $2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$ , it may be computed. When false positives and false negatives are equally relevant, the  $F_1$  score is helpful.

**Fig. 5** Recall (sensitivity) of data-driven decision support system in e-governance



**Fig. 6**  $F_1$  scores for a data-driven decision support system in e-governance



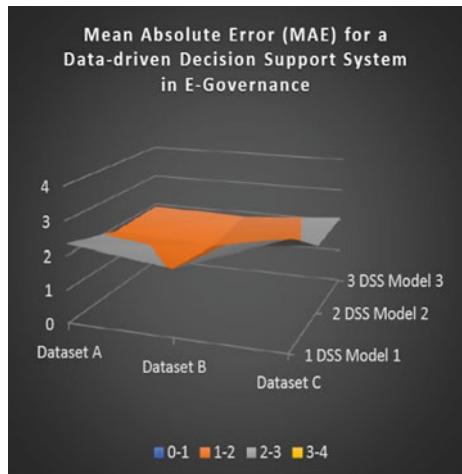
#### 4.5 Mean Absolute Error (MAE)

The MAE measure may be applied to DSSs that call for the prediction of numerical values. In order to compare the projected and actual values, the average absolute difference is calculated. Better accuracy in forecasting quantitative outcomes is indicated by a lower MAE.

The average absolute difference between the predicted values and the actual values for each DSS model on each dataset is represented by the MAE values in Table 4 (see below). The absolute value of the difference between the expected and actual values is taken, and then averaged, to determine the MAE. Table 4 enables comparisons

**Table 4** Mean absolute error (MAE) for a data-driven decision support system in e-governance

S. No.	System	Dataset A	Dataset B	Dataset C
1	DSS model 1	2.35	1.98	3.12
2	DSS model 2	1.87	1.72	2.04
3	DSS model 3	1.92	1.85	2.1

**Fig. 7** Mean absolute error (MAE) for a data-driven decision support system in e-governance

between the MAE values of the DSS models across various datasets. Figure 7 displays the findings for mean absolute error (MAE) for an e-government data-driven decision support system.

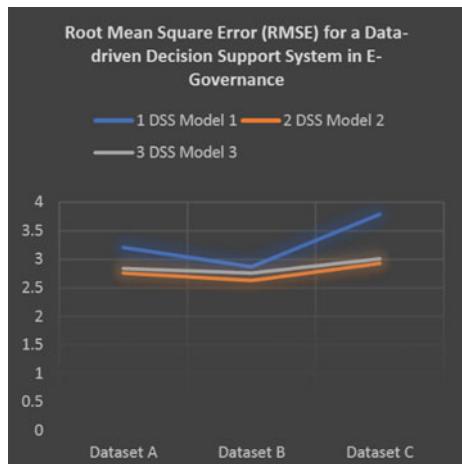
#### 4.6 Root Mean Square Error (RMSE)

RMSE is a statistic that is utilized for quantitative predictions and is similar to MAE. The average squared difference between the values that were predicted and those that were observed is calculated. An improved ability to forecast quantitative events is indicated by a decreased RMSE, similar to MAE.

The given Table 5 gives the results of the evaluation of three distinct DSS models using datasets A, B, and C. For each DSS model on each dataset, the RMSE values are the square root of the average squared difference between the predicted values and the actual values. When calculating the RMSE, the average of the squared differences' square root is used. The graph in the table enables a comparison of the DSS models' RMSE values across various datasets. Figure 8 illustrates the findings for the root mean square error (RMSE) for a data-driven decision support system in e-government.

**Table 5** Root mean square error (RMSE) for a data- driven decision support system in e-governance

S. No.	System	Dataset A	Dataset B	Dataset C
1	DSS model 1	3.21	2.87	3.79
2	DSS model 2	2.76	2.63	2.93
3	DSS model 3	2.84	2.76	3.01

**Fig. 8** Root mean square error (RMSE) for a data-driven decision support system in e-governance

## 5 Conclusions

The experimental results frequently involve a contrast between AI-driven decision support systems and more traditional approaches to assessing how well they function in terms of impact, effectiveness, and efficiency. Quantitative measurements, such as accuracy, precision, recall, or other performance indicators are included in the results to evaluate the system's ability to provide actionable insights, analyze data, and support policy decisions. The results of data-driven decision support systems in e-governance contribute to the body of knowledge in the field by providing insightful information on how well AI-driven decision support systems facilitate policymaking within the context of e-governance. With an emphasis on their capability to analyze and comprehend data for policymaking, the following quantitative metrics are utilized to evaluate the performance of data-driven decision support systems (DSS) in e-governance.

## References

1. Alam N et al (2017) Internet of Things: a literature review. In: 2017 Recent developments in control, automation & power engineering (RDCAPE) (2017), pp 192–197
2. Krishnam NP et al (2022) Analysis of current trends, advances and challenges of machine learning (ML) and knowledge extraction: from ML to explainable AI. Ind Qualifications Inst Adm Manage UK 58 (2022) 54–62
3. Verma M et al (2022) Experimental analysis of geopolymmer concrete: a sustainable and economic concrete using the cost estimation model. *Adv Mater Sci Eng*
4. Vats P (2016) A comprehensive review of cyber terrorism in the current scenario. In: 2016 Second international innovative applications of computational intelligence on power, energy and controls with their impact on humanity (CIPECH), pp 277–281
5. Manju M et al (2020) A comprehensive literature review of penetration testing & its applications. In: 2020 8th International conference on reliability, infocom technologies and optimization (trends and future directions) (ICRITO). IEEE
6. Gosai A et al (2014) A comparative analysis of various cluster detection techniques for data mining. In: 2014 international conference on electronic systems, signal processing and computing technologies. IEEE
7. Singh S et al (2022) A novel approach for implementation of software requirement specifications using the humpback whale optimization model. In: ICT systems and sustainability: proceedings of ICT4SD 2022. Springer Nature Singapore, Singapore, pp 123–132
8. Sharma N et al (2022) A robust framework for governing blockchain-based distributed ledgers during COVID-19 for academic establishments. In: ICT with intelligent applications: proceedings of ICTIS 2022, vol 1. Springer Nature Singapore, Singapore, pp 35–41
9. Sharma AK et al (2022) Deep learning and machine intelligence for operational management of strategic planning. In: Proceedings of third international conference on computing, communications, and cyber-security: IC4S 2021. Springer Nature Singapore, Singapore
10. Vats P et al (2022) A hybrid approach for retrieving geographic information in wireless environment using indexing technique. *ICT Analysis and Applications*. Springer Singapore
11. Qureshi A et al (2022) A review of machine learning (ML) in the internet of medical things (IOMT) in the construction of a smart healthcare structure. *J Algebraic Stat* 13(2):225–231
12. Vats P, Madan S, Gosain A (2013) A comparative study of various object-oriented testing techniques. In: 2013 Fourth international conference on computing, communications and networking technologies (ICCCNT). IEEE, pp 1–8
13. Kaushik H et al (2022) Deployment and layout of deep learning-based smart eyewear applications platform for vision disabled individuals. *J Positive Sch Psychol* 4167–4173
14. Doja F et al (2022) A comprehensive framework for the IoT-based smart home automation using Blynk. In: Information and communication technology for competitive strategies (ICTCS 2021) intelligent strategies for ICT. Springer Nature Singapore, Singapore, pp 49–58
15. Kaushik S et al (2022) A comprehensive analysis of mixed reality visual displays in context of its applicability in IoT. In: 2022 International mobile and embedded technology conference (MECON). IEEE
16. Varshney S et al (2022) A blockchain-based framework for IoT based secure identity management. In: 2022 2nd international conference on innovative practices in technology and management (ICIPTM), vol 2. IEEE
17. Aalam Z et al (2022) A comprehensive analysis of testing efforts using the avisar testing tool for object-oriented software's. In: Intelligent sustainable systems: selected papers of WorldS4 2021, vol 2. Springer Singapore
18. Vats P, Saha A (2019) An overview of SQL injection attacks. Available at SSRN 3479001
19. Kaur R et al (2021) Literature survey for IoT-based smart home automation: a comparative analysis. In: 2021 9th International conference on reliability, infocom technologies and optimization (trends and future directions) (ICRITO). IEEE

20. Chauhan K et al (2022) A comparative study of various wireless network optimization techniques. In: Information and communication technology for competitive strategies (ICTCS 2020) ICT: applications and social interfaces. Springer Singapore, 2022
21. Gupta PNSA (2015) Expansion of existing use cases using extend relationship. IJCSNS 15(9):44
22. Vats P et al (2014) A novel study of fuzzy clustering algorithms for their applications in various domains. In: The 4th joint international conference on information and communication technology, electronic and electrical engineering (JCTEE). IEEE
23. Butterworth M (2018) The ICO and artificial intelligence: the role of fairness in the GDPR framework. Comput Law Secur Rev 34(2):257–268
24. Casares AP (2018) The brain of the future and the viability of democratic governance: the role of artificial intelligence, cognitive machines, and viable systems. Futures 103:5–16
25. Čerka P, Grigienė J, Sirbikytė G (2017) Is it possible to grant legal personality to artificial intelligence software systems? Comput Law Secur Rev 33(5):685–699
26. Janssen, Marijn, et al (2020) Data governance: organizing data for trustworthy artificial intelligence. Gov Inf Quart 37(3):101493
27. Kouziokas GN (2017) The application of artificial intelligence in public administration for forecasting high crime risk transportation areas in urban environment. Transportation research procedia 24 (2017): 467–473.
28. Ku C-H, Leroy G (2014) A decision support system: Automated crime report analysis and classification for e-government. Gov Inf Q 31(4):534–544
29. Liu, Shuhua Monica, and Yushim Kim. “Special issue on internet plus government: New opportunities to solve public problems?” Government information quarterly 35.1 (2018): 88–97.
30. Matheus R, Janssen M, Maheshwari D (2020) Data science empowering the public: data-driven dashboards for transparent and accountable decision-making in smart cities. Gov Inf Q 37(3):101284
31. Sun TQ, Medaglia R (2019) Mapping the challenges of artificial Intelligence in the public sector: Evidence from public healthcare. Gov Inf Q 36(2):368–383
32. Wirtz BW, Weyerer JC, Kehl I (2022) Governance of artificial intelligence: a risk and guideline-based integrative framework. Gov Inf Q 39(4):101685
33. Brynjolfsson E, Mitchell T (2017) What can machine learning do? Workforce implications. Science 358(6370):1530–1534
34. Xia J et al (2014) Development of a GIS-based decision support system for diagnosis of river system health and restoration. Water 6(10):3136–3151
35. Mandot M et al (2014) A comparative study of genetic algorithms for its applications in object oriented testing. In: 2014 International conference on issues and challenges in intelligent computing techniques (ICICT). IEEE
36. Jain D et al (2022) A comprehensive framework for IoT-based data protection in blockchain system. In: Information and communication technology for competitive strategies (ICTCS 2021) intelligent strategies for ICT. Springer Nature Singapore, Singapore, 473–483
37. Gossain A, Mandot M (2020) SARLA-A 3-tier architectural framework based on the ACO for the probabilistic analysis of the regression test case selection and their prioritization. In: 2020 8th international conference on reliability, infocom technologies and optimization (Trends and future directions) (ICRITO). IEEE
38. Alam Z et al (2021) A multi-factorial code coverage based test case selection and prioritization for object oriented programs. In: ICT systems and sustainability: proceedings of ICT4SD 2020, vol 1. Springer Singapore
39. Neha K et al (2017) AVINASH—A three tier architectural metric suit for the effort estimation in testing of OOS. In: 2017 International conference on intelligent communication and computational techniques (ICCT). IEEE
40. Ali I et al (2017) E-Governance: a study of issues & challenges in Indian context. Int J Eng Manage Res (IJEMR) 7(3):664–667

41. Gulati R et al (2014) A literature review of Bee Colony optimization algorithms. In: 2014 Innovative applications of computational intelligence on power, energy and controls with their impact on humanity (CIPECH) (2014), pp 499–504
42. Mandot P et al (2014) A comparative analysis of ant colony optimization for its applications into software testing. In: 2014 Innovative applications of computational intelligence on power, energy and controls with their impact on humanity (CIPECH). IEEE
43. Kaur R et al (2022) A comprehensive approach for recognizing the ocular impression using machine learning-based CNN and LBP plainer interpolation. In: ICT Infrastructure and computing: proceedings of ICT4SD 2022. Singapore: Springer Nature Singapore, 2022, pp 721–728
44. Jain E et al (2022) A CNN-Based neural network for tumor detection using cellular pathological imaging for lobular carcinoma. In: ICT with intelligent applications: proceedings of ICTIS 2022, vol 1. Springer Nature Singapore, Singapore, 541–551
45. Kapula PR et al (2022) The block chain technology to protect data access using intelligent contracts mechanism security framework for 5g networks. In: 2022 2nd International conference on advance computing and innovative technologies in engineering (ICACITE). IEEE
46. Gupta A et al (2022) A sustainable green approach to the virtualized environment in cloud computing. In: Smart trends in computing and communications: proceedings of SmartCom 2022. Springer Nature Singapore, Singapore, pp 751–760
47. Sharma AK et al (2022) An IoT-Based temperature measurement platform for a real-time environment using LM35. In: Information and communication technology for competitive strategies (ICTCS 2021) ICT: applications and social interfaces. Springer Nature Singapore, Singapore, pp 603–612
48. Singh P et al (2022) Cloud-Based patient health information exchange system using blockchain technology. In: Information and communication technology for competitive strategies (ICTCS 2021) intelligent strategies for ICT. Springer Nature Singapore, Singapore, pp 569–577
49. Garg RK, Saini M, Vats P (2019) A novel study of application of information & communication technology in library classification. Available at SSRN 3478986
50. Phogat M et al (2022) Identification of MRI-Based adenocarcinoma tumors with 3-D coevolutionary system. In: Information and communication technology for competitive strategies (ICTCS 2021) intelligent strategies for ICT. Springer Nature Singapore, Singapore, pp 587–597
51. Vats P, Biswas SS (2023) Big data analytics in real time for enterprise applications to produce useful intelligence. Data Wrangling Concepts Appl Tools 187
52. Upreti K et al (2023) OFDA: a comprehensive and integrated approach for predicting estimated delivery time for online food delivery. In: Intelligent sustainable systems: selected papers of WorldS4 2022, vol 2. Springer Nature Singapore, Singapore, pp 325–333
53. Arora A et al (2023) OCD: on-demand ordering food through online crowdsourcing. In: Intelligent sustainable systems: selected papers of WorldS4 2022, vol 2. Springer Nature Singapore, Singapore, pp 539–548
54. Bhagat AD et al (2022) A survey of cloud architectures: confidentiality, contemporary state, and future challenges. In: 2022 3rd International conference on issues and challenges in intelligent computing techniques (ICICT). IEEE
55. Upreti K et al (2022) A Comprehensive framework for online job portals for job recommendation strategies using machine learning techniques. In: ICT infrastructure and computing: proceedings of ICT4SD 2022. Springer Nature Singapore, Singapore, pp 729–738
56. Arora A et al (2022) A comprehensive study on social network analysis for digital platforms to examine and solve the behavioral patterns of everyday routines. In: ICT systems and sustainability: proceedings of ICT4SD 2022. Springer Nature Singapore, Singapore, pp 13–21

# The Infrastructure Development of Contemporary Medical Devices Based on Internet of Things Technology



Haider Al-Kanan and Ahmed S. Alzuhairi

**Abstract** Internet of Things (IoT) is a modern technology that has significantly improved efficiency and deployment of medical devices in healthcare sector. Various approaches have been presented in the literature for integrating IoT with medical devices to solve various healthcare problems and overcome the potential challenges in conventional approaches, such as enhancing device's operational efficiency, reducing implementational costs, and improving both access and patient's safety. This paper highlights different implementation approaches based on both business and technical goals which are considered very important steps and strategy analysis in developing IoT medical devices. These goals enable the developers during planning phase to select the appropriate standards. This is because there are many different industry standards that can be adopted for integrating IoT with medical devices as discussed in this paper, e.g., communication standards for connectivity, artificial intelligence algorithms for data analysis, and cloud computing have various implementation requirements. Finally, the networking technical goals of IoT are discussed for specifying the reliability and efficiency of the developed IoT approach.

**Keywords** Internet of Things · Medical devices · Healthcare systems · Technology development

## 1 Introduction

The ongoing technology evolution in deploying Internet of Things (IoT) becomes more popular in various sectors of medicine. This rapid progress has made significant growth and a big contribution to revenue and work. A few years ago, the diagnosis of and abnormality in the human body can only after undergoing a physical analysis at the hospital; furthermore, most of patients must stay in hospital during the treatment stage. This has resulted in cost increase as well as put a strain on health facilities in

---

H. Al-Kanan (✉) · A. S. Alzuhairi

Department of Medical Instruments Technology, Al-Kut University College, Alhay, Wasit 52011, Iraq

e-mail: [alkanan.haider@gmail.com](mailto:alkanan.haider@gmail.com)

rural and remote locations. Technological progress achieved during recent years has allowed the diagnosis of various diseases and health monitoring using miniaturization devices such as smart watches. In addition, technology has transformed the hospital-centered healthcare system into patient-centered system [1–12].

The COVID-19 pandemic has accelerated the use of telemedicine and related technologies with rapid adoption of medical IoT technologies such as remote patient monitoring and medical robotics, opening vast digitization opportunities for IoT solution providers as depicted in Fig. 1 [6]. For instance, with IoT medical devices, doctors can monitor a patient's health status and fitness. Physiological medical devices and methods are easily traceable for monitoring the patient's condition which is helpful to avoid early health problems and mitigating potential of sickness progress. The number of connected medical devices are expected to continue growing due to an efficient development of both hardware and software technologies, such as digital healthcare apps allow patients to schedule appointments without having to call a doctor's office and wait for a receptionist. Furthermore, medical information technology allows doctors to carry information wherever they go through apps on their smartphones [13–15].

The IoT improved the customer experience and decision-making when combined with emerging technologies such as 5G communications, cloud computing, and machine learning. This integration facilitates reliable sharing information which is analyzed remotely and decisions are made automatically with help of a modern protocol and algorithms. The applications of IoT can go beyond the healthcare systems to be found in agriculture, automobile, family, and health care. The growing popularity of IoT is due to its advantages including higher display accuracy, lower cost, and better predictability of future events [14–18].

Nowadays, most IoT systems use a GUI that acts as a dashboard for healthcare providers and implement user controls, data storage, and visualization. Data storage and accessibility have become significant challenge in some implementation scenarios because it plays an important role in IoT system as a large amount of data is collected/recorded from many types of sources (sensors, mobile phones, e-mail, software, and applications). These types of data must be made available to physicians, caregivers, and authorized parties through the cloud/server allowing for quick patient diagnosis and medical intervention if necessary [16].

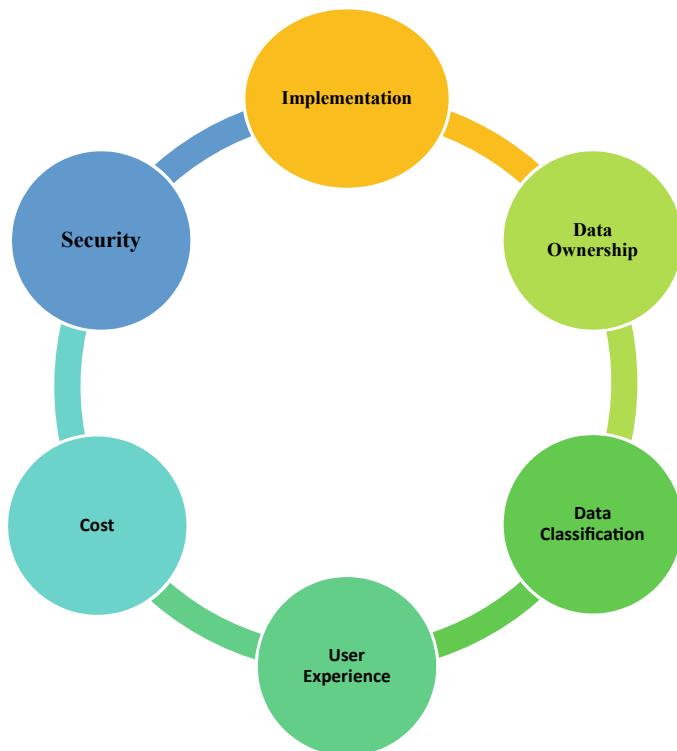
IoT developers in healthcare sectors have a tremendous freedom when proposing solution for integration of different medical devices, this is because there is no specified industry/medicine standard. However, most of the ongoing approaches have been developed based on the existing cutting-edge technologies and infrastructures. With the aim of maximizing the usability of IoT in health system, many countries have adopted new technologies such as 5G communications and medical policies. The motivation of this research is to review the advancement of adopting IoT technology related to health systems, and to provide a foundation of the integration between the traditional medical instruments and the adopted approaches in terms of cloud computing, connectivity, network technical goals, and standards.

## 2 IoT Infrastructure in Health Care

Although different design solutions can be used in IoT medical devices, the main point is to enable reliable integration components of information technology (IT), such as networking and computing. These components represent the infrastructure of developing any approach of IoT medical devices.

Before choosing IoT infrastructures, it is very important to analyze both business goal and technical goal of the integrated medical instruments. In addition, the design solution depends on the other aspects such as the applications and services, e.g., some diseases require a complex process of care strategy, and the topology must follow medical policies and diagnoses procedure. Business constraints are the early steps of defining the specifications of the IoT solution in healthcare industry as depicted in Fig. 1 and described as follows [18–24]:

- Implementation: The deployment and operation of IoT for medical devices must satisfy full compliance of used devices with both industry and medical standards.

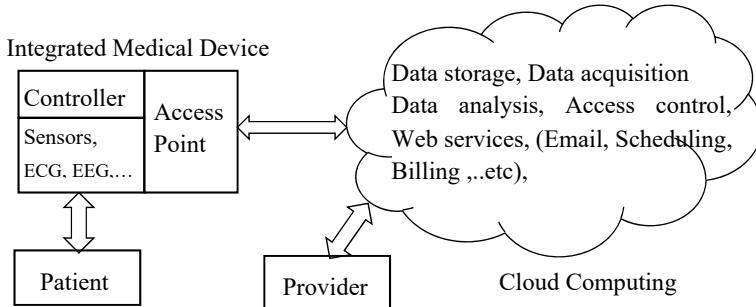


**Fig. 1** Business framework goals for IoT applications

- Security: The health information records are subject to significant privacy regulations and security risks in addition to the compliance with health insurance disclosure and accountability act. Most of IoT medical devices use wireless communication; therefore, the transmitted data exposes to a wide range of cybersecurity threats and fraud attempts such as illegally obtaining medications and other medical services.
- Data rights: The data factor includes patients, software vendors, and other health-care providers who are authorized to create or touch data during its lifecycle. The ownership of the data depending on the context, each party's data rights can be complex.
- Expenses: This includes two types of cost; during the initial implementation phase and other costs related to product deployment and operation for achieving a positive return on long-term investment.
- User experience: Due to the different range of user's knowledge as they might belong to various culture and background, it can be difficult to design medical devices and applications that are easy for patients to use. Poorly designed devices lead to limited or faulty data collection when patients remove the device or use it improperly.
- Data classification: Different types of data are typically evolved during the deployment and operation of specific IoT application, e.g., emerging health data is health information inferred by artificial intelligence (AI) applications from non-health related data. AI technology can examine consumer tracks and other data and turn them into medical data. Another example of AI-IoT integration includes location tracking approach for predicting the spread of infectious diseases.

The technical design standards facilitate the integration and deployment of various IoT medical devices to ensure reliability, safety, and operational compatibility.

Various strategies and operational protocols have been adopted in medical devices for diagnoses, treatment, and monitoring patient's condition. A unified IoT medical approach is widely deployed in industry as depicted in Fig. 2 which consists of medical instruments attached or close to the patient under health care for real-time monitoring of different human tissues. The sensors collect measurement such as blood pressure, temperature, ECG, EEG, and many other recordings. The controller is typically used to configure the sensor settings and some controllers are used for other several purposes such as collecting tissue samples and remote surgery. The access point is used for connectivity and interface with cloud. The cloud computing is a core side which is used for many functions such as data storage and analytics, and communication media between health providers and patients.



**Fig. 2** Unified solution of IoT for healthcare industry

### 3 IoT Wireless Standards

The communication standards various in IoT connected devices depending on the application and the operational costs. For example, remote surgery IoT applications require low latency and high data rate communications such as 5G because the time factor is extremely sensitive, other applications such as wearable sensors require low-power and low-speed communications to transmit few kilobyte payload data which is sufficient to use low-cost Bluetooth or ZigBee protocols. In addition, radio frequency identifications (RFID) and Wi-Fi are the commonly most deployed wireless technology nowadays in IoT integration of medical devices [13]. The IoT developers often classify the wireless communication to short-range, long-range, and medium range technologies, e.g., RFID is a short-range standard that is used in smart sensors and smartwatches which can be operated in passive mode to cancel-out the needs of external power sources. On the other hand, the Wi-Fi technology offers longer communication range of up to 100 m. Furthermore, satellite and cellular communications are used when deploying IoT devices over very long-distance because it offers reliable wireless coverage with significant advantages such as high-speed date rate, stability, and it offers secure encrypted protocols [14].

The connectivity among the IoT devices is an important implementation aspect that ensures all the application objects are functioning well during different environmental conditions and circumstances. Some network topology offers redundant connections in case the communication link is down. However, this approach might be costly due to the implementation complexity. A point-to-point connectivity is the simplest network topology in healthcare systems as well as personal-area-network (PAN) for short-range communication, local-area-network (LAN) for medium range, and wide-area-network (WAN) for long-range. In addition, the IoT nodes can be connected using certain structure such as ring, star, mesh, tree, or hybrid.

Although the IoT innovations in health care show the potential to enhance patient health and obtaining medical services, most of these IoT devices pose cybersecurity risks that can adversely affect the overall outcomes. This is because medical devices typically collect and process highly sensitive personal health data. Key requirements

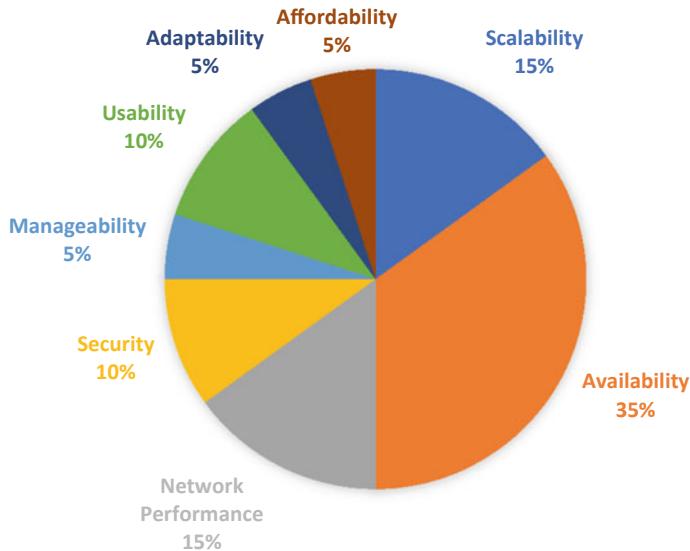
for an efficient security approach must be implemented on both device level and data security layer, including device authentication, integrity, and data confidentiality. Various security approaches have been used in the IoT medical devices depending on the application and implementation complexity, but most of the developed solutions concentrate on the communication level such as encryption techniques associated with the communication to protect user privacy during operational phase, e.g., the elliptic-curve cryptography (ECC) is more relevant and lower complexity approach that has been widely deployed to meet the security compliance requirements [12–15].

## 4 IoT Network Architecture

Network designers and conventional users often employ many different terms to describe both of business and technical goals in the healthcare development and other marketing aspects. The business goals play significant roles to achieve the plan requirements which is beyond the scope of this work. On the other hand, technical goals define the network specifications and architecture to meet the customer expectations. Common technical goals include security, availability, network performance, scalability, manageability, affordability, adaptability, and usability [27]. It is very difficult to meet all these technical goals at 100% in any network architecture because these goals come with trade-offs. For instance, meeting network performance requirements can make it difficult to achieve the affordability goals. An example of IoT network goal is depicted in Fig. 3 which shows a feasible implementation approach that can be used in IoT network of medical instruments. In this example, some of the goals are critical requirements to users, such as the availability criteria of the network medical services which is worth 35% from the overall technical goals. Furthermore, the scalability and network performance worth 15% each, which represent the second and more important criteria for reliable operation and future network expansion. The other technical categories are applications and marketing dependents and subject to the financial budget. Therefore, 5–10% could be sufficient criteria for most IoT medical applications.

### 4.1 Availability

The availability is an important and critical goal for the both patient and provider, because it refers to how long the IoT network is up and running. The availability is subject to many uncontrolled factors such as natural disasters such as floods, fires, hurricanes, and earthquakes, in addition to device failures. Device and network redundancy are often adopted approaches to account for such concerns and to meet such as five nines (i.e., 99.999) availability goal.



**Fig. 3** IoT network specifications depict the overall consideration of technical goals

## 4.2 Scalability

Scalability refers to how much expansion the network design must support. Scalability is a major goal for many customers designing enterprise networks in healthcare sector. The IoT network developers must analyze the future goals and understand how their networks will grow in the next years, such as by adding new devices, users, applications, and external network connections. An efficient network architecture should scale as the network usage and size increase. The scalability requirements of IoT network are typically achieved by selecting an appropriate technology, network topology and protocols.

## 4.3 Network Performance

Network performance goal is another important technical aspect in different applications of IoT of healthcare industry. The network performance is typically characterized by throughput, latency, utilization, and bandwidth.

#### ***4.4 Manageability***

Each IoT application of medical devices has different network management goals which are addressed in the early plan of the proposed project. The manageability goals include accounting management, security management, performance management, configuration management, and fault management.

#### ***4.5 Affordability***

Low cost is often the primary goal for patients when using IoT network related to diagnoses, monitoring, scheduling, and other usage purposes. On the other hand, the network designer considers the budget constraint as a factor when selecting the hardware/software technology.

#### ***4.6 Adaptability***

The network adaptability refers to how quickly connected devices should adapt for certain changes and upgrades. A reliable IoT network design should adapt to the changing of traffic patterns and QoS requirements as well as the system upgrade including new technology and protocols. Other adaptability goals can come in the form of changing health regulatory requirements and policies.

#### ***4.7 Usability***

Smart IoT devices are primarily developed for monitoring different medical devices including measurement accuracy.

### **5 Cloud Computing in IoT**

Artificial intelligence (AI) technique is widely used to understand certain data behavior and sometimes to auto-responding for controlling parts of the IoT system. The AI technology shows significant improvement when designing smart IoT medical devices. For instance, AI can implement more secure connectivity, reduce communication latency, and decrease the needs of storing redundant or unnecessary collected data. In addition, AI can address the security and privacy challenges because it enables encryption/decryption solutions [6, 7].

The AI computation can be implemented on each medical device as a part of the IoT system. However, this approach is costly and inefficient because this requires independent processing unit which poses a relatively high-power consumption due to the AI complex computations. Therefore, most of the IoT healthcare systems shift data processing in the clouds because the cloud system usually has more efficient and reliable systems dedicated for such purposes and it is accessible from different locations, such as Microsoft Azure, Amazon Web Services, and Google Cloud Platform. Another solution has been widely adopted nowadays in industry by shifting the data computation to the edge of the network in order to decrease the latency for critical and time-sensitive applications.

## 6 Contemporary Applications of IoT

IoT is an important backbone for utilizing the operations of medical devices in health care and to revolutionize entire industries. The IoT can help change the way health systems work and the way they deliver care in an efficient way. The transitioning from the traditional medical instruments to smart and an efficient approach is the ongoing trend in healthcare development from different possible angles [5–11]. Some of the most popular IoT solutions include wearables systems, home monitoring, and sensors. Furthermore, some of the contemporary solutions and applications that allow improvement of the patients/providers experience in different healthcare environments are presented as follows.

### 6.1 Smart IoT Sensors

Smart IoT devices are primarily developed for monitoring different medical devices including measurement accuracy, performance, and other environmental conditions, e.g., for routine maintenance and repair activity on medical equipment, predefined thresholds are set to alert the responsible personals for taking the required actions. The notification technique has a potential to reduce the overall cost required for repairing faulty devices and ensuring that medical instruments performing optimal load performance. Furthermore, optimizing equipment deployment can help patients get the information they need to provide them with high-quality medical care.

### 6.2 Smart Pill Containers

Smart pillboxes give individuals the same kind of IoT inventory management and product tracking capabilities that major manufacturers employ in their stores and production facilities. Using the data transfer capabilities of IoT devices, smart

container can help patients refill running-out prescriptions, send notifications, and provide medical assistance [5].

### **6.3 Smart Hospital Beds**

Smart beds technology has recently shown significant roles when dealing with high contiguous diseases such as COVID-19 pandemic. Smart beds are equipped with IoT that allows remote control and monitoring system for tracking patients' conditions. In addition, the IoT system can decrease the required time to respond in case of emergency circumstances, e.g., it helps the medical staff finding free beds as soon as they become available, and make the work of nurses and other healthcare professionals more efficient and easier. Furthermore, IoT allows integrating some tools on the bed such as a blood glucose monitoring which alerts medical staff to any issues that need to be addressed [13, 14].

### **6.4 Temperature Sensors**

Different types of temperature sensors are recently integrated with IoT and they have become essential for real-time and accurate tracking of the temperature variation over time. Furthermore, smart signal analysis approaches are used to detect or predict a pattern during monitoring patients' temperature to alert the medical staff and other purposes such as maintaining specific environmental conditions in the medical facilities, e.g., the infrared temperature sensors are widely used instruments for monitoring the laboratory, medications, and body tissues.

### **6.5 Wearable Devices**

The field of wearable technology has significantly exploded in recent years using many different approaches in health care, e.g., smart belt technology is one of the state-of-the-art wearable technology revolutions that can help patients find the track and alert emergency facilities in need of assistance. This technology also helps seniors who want to remain calm while living with family and friends. Apple watch is another smart application in health care which integrates several various sensors to collect and track valuable information such as blood sugar, heart beats, active minutes, movement pattern and other biometric information which can be used for improving both medical concern and physical activities.

## 6.6 Future of IoT Medical Instruments

IoT integration with medical instrumentations is rapidly developing due to the ongoing improvements in supporting technologies, such as next generation wireless communication, nanotechnology, and intelligent computing. The contemporary IoT medical applications include robotics surgery, the implantable cardiac instrumentation, smart insulin pumps, vision and hearing aids equipment, and many other types of medical instrumentations [25–32].

## 7 Conclusions

IoT is an attractive technology, and its potential to improve health care remains endless. This article has discussed the implementation of IoT medical instrumentations from the perspective of business goals and technical standards. In fact, the IoT is open-source technology without unique international standard which often poses flexibility in selecting the appropriate instrumentations, however, those devices must satisfy full compliance of industry standards such as the connectivity protocols and data analysis as discussed in this paper. IoT network architecture plays an important role in connecting various medical instrumentations. Hence, this paper has presented different criteria for evaluating the overall IoT medical approach in terms of model performance, usability, efficiency, security, adaptability, affordability, scalability, availability, and manageability.

## References

1. Mukati N, Namdev N, Dilip R et al (2023) Healthcare assistance to COVID-19 patient using Internet of Things (IoT) enabled technologies. Mater Today Proc 80(Part 3):3777–3781
2. Bajaj V, Ansari I (2023) Internet of Things in biomedical sciences, challenges and applications, Indian institute of information technology, design and manufacturing. IOP publishing Ltd India
3. Griggs K, Ossipova O, Kohlios C et al (2018) Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. J Med Syst 42(7):130
4. Mamun A, Alam M, Hasan Z et al (2023) IoT-Based smart health monitoring system: design, development, and implementation. In: The fourth industrial revolution and beyond. Lecture notes in electrical engineering, vol 980. Springer
5. Sahu D, Pradhan B, Wilczynski S et al (2023) 6-Development of an internet of things (IoT)-based pill monitoring device for geriatric patients. Adv Methods Biomed Signal Process Anal 129–158
6. Ding X, Zhang Y, Li J et al (2023) A feasibility study of multi-mode intelligent fusion medical data transmission technology of industrial Internet of Things combined with medical Internet of Things. Internet of Things 21
7. Rodríguez E, Otero B, Canal R (2023) A survey of machine and deep learning methods for privacy protection in the Internet of Things. Sensors 23:1252

8. Kranzfelder M, Zywitz D, Jell T et al (2012) Real-time monitoring for detection of retained surgical sponges and team motion in the surgical operation room using radio-frequency-identification (RFID) technology: a preclinical evaluation. *J Surg Res* 175:191–198
9. Mathew P (2018) Applications of IoT in healthcare. In: Cognitive computing for big data systems over IoT. Springer, pp 263–288
10. Jagadeeswari V (2018) A study on medical Internet of Things and big data in personalized healthcare system. *Health Inf Sci Syst* 6:14
11. Karthick R, Prabaharan A, Selvaprasanth P (2019) Internet of things based high security border surveillance strategy. *Asian J Appl Sci Technol (AJAST)* 3:94–100
12. Al-Kashoash H, Al-Nidawi Y, Kemp A (2016) Congestion-aware RPL for 6LoWPAN networks. In: Wireless telecommunications symposium (WTS), pp 1–6
13. Pradhan B, Bhattacharyya S, Pal K (2021) IoT-Based applications in healthcare devices. *J Health Eng* 2021(18):2021
14. Karthick R, Ramkumar R, Akram M et al (2021) Overcome the challenges in bio-medical instruments using IOT—a review. *Mater Today: Proc* 45(Part 2):1614–1619
15. Subhan F, Mirza A, Su'ud M et al (2023) AI-Enabled wearable medical internet of things in healthcare system: a survey applications. *Science* 13:1394
16. Peng H, Tian Y, Kurths J et al (2017) Secure and energy-efficient data transmission system based on chaotic compressive sensing in body-to-body networks. *IEEE Trans Biomed Circuits Syst* 11(3):558–573
17. Shiny M, Ramrao N, Murugan K (2023) A review on ongoing medical care observing framework for cardiovascular patients utilizing IoT. In: 7th International conference on computing methodologies and communication (ICCMC), Erode, India, 2023, pp 1310–1318
18. Gatouillat A, Badr Y, Massot B et al (2018) Internet of medical things: a review of recent contributions dealing with cyber-physical systems in medicine. *IEEE Internet of Things J* 5(5):3810–3822
19. Dang L, Piran M, Han D et al (2019) A survey on internet of things and cloud computing for healthcare. *Electronics* 8(7):768
20. Karthick R, Sundararajan M (2017) A reconfigurable method for time correlated MIMO channels with a decision feedback receiver. *Int J Appl Eng Res* 12
21. Dohr A, Modre-Osprian R, Drobics M et al (2010) The Internet of things for ambient assisted living. *ITNG* 10:804–809
22. Balasundaram A, Routray S, Prabu AV et al (2023) Internet of things (IoT) based smart healthcare system for efficient diagnostics of health parameters of patients in emergency care. *IEEE Internet of Things J.* <https://doi.org/10.1109/IJOT.2023.3246065>
23. Khalique A, Singh K, Sood S (2010) Implementation of elliptic curve digital signature algorithm. *Int J Comput Appl* 2(2):21–27
24. Yamashita K, Iwakami Y, Imaizumi K et al (2008) Identification of information surgical instrument by ceramic RFID tag. In: 2008 World automation congress, pp 1–6
25. Ahram T, Sargolzaei A, Sargolzaei S et al (2017) Blockchain technology innovations. In: 2017 IEEE technology & engineering management conference (TEMSCON), pp 137–141
26. Dinis H, Zamith M, Mendes P (2015) Performance assessment of an RFID system for automatic surgical sponge detection in a surgery room. In: Conference proceedings: annual international conference of the IEEE engineering in medicine, pp 3149–3152
27. Oppenheimer P (2004) Top-Down network design, 2nd edn. Cisco Press. ISBN: 1587051524
28. Pandey S, Vanshika A, Dwivedi R (2023) A secure design of healthcare system with blockchain and Internet of Things (IoT). In: 2023 International conference on intelligent data communication technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp 105–111. <https://doi.org/10.1109/IDCIoT56793.2023.10053491>
29. Czekster RM, Grace P, Marcon C, Hessel C et al (2023) Challenges and opportunities for conducting dynamic risk assessments in medical IoT. *Appl Sci* 13:7406. <https://doi.org/10.3390/app13137406>
30. Naeem S, Ali A, Memon K et al (2023) A review of flexible high-performance supercapacitors for the internet of things (IoT) and artificial intelligence (ai) applications. *Energ Thermofluids Eng* 3:1–9

31. Selmani et al (2023) Design of patient monitoring data node on the way of implementing medical 4.0. In: 2023 3rd International conference on innovative research in applied science, engineering and technology (IRASET), Mohammedia, Morocco, pp 1–5
32. Saravanan C, Krishnamoorthy NV, Vanakovarayan S et al (2023) Design and implementation of patient monitoring system based on IoT using oxygen saturation. In: Eighth international conference on science technology engineering and mathematics (ICONSTEM), Chennai, India, 2023, pp 1–5

# Business Intelligence System Adoption Project in the Area of Investments in Financial Assets



Beata Dratwińska-Kania and Aleksandra Ferens

**Abstract** Organizations often adopt business intelligence (BI) systems to improve the standardization and speed of collecting multidimensional information related to their business activity. This study highlights the benefits of implementing a BI system and presents an original project that transforms a traditional information system into a BI system, focusing on investments in financial assets in a hypothetical enterprise. The authors believe this project is unique and valuable because no similar study has been conducted before. The study examines the necessary data using proposals of OLAP cubes and demonstrates the advantages of implementing a BI system, although the research is limited to investments in financial assets due to the complexity of building a comprehensive BI system. The information obtained from the BI system project will aid managerial staff in making effective decisions, which is essential for organizational competitiveness and survival. The study aims to describe the construction and benefits of BI systems and present an original, simplified project of the BI system in an organization related to investments in financial assets. The research methods include a critical literature analysis and a case study. The study provides initial confirmation of the research hypothesis that implementing a BI system positively impacts decision-making processes related to investments in financial assets.

**Keywords** Business intelligence · Financial assets · OLAP cubes

## 1 Introduction

In today's business world, managing a successful enterprise relies heavily on data from various sources that are often incompatible with each other. This presents a significant challenge for decision-makers. As a result, entrepreneurs are seeking IT solutions to assist with decision-making processes. One popular solution is business intelligence (BI) systems, which organizations purchase and customize to meet their

---

B. Dratwińska-Kania () · A. Ferens

Faculty of Finance, University of Economics, ul. 1 Maja 50, 40-287 Katowice, Poland  
e-mail: [beatakania@ue.katowice.pl](mailto:beatakania@ue.katowice.pl)

specific needs. By embracing new technologies, organizations can gain a competitive edge and increase the overall value of their enterprise. BI systems allow management to analyze large datasets quickly, providing new insights that contribute to more efficient decision-making.

This article explores the question of how to improve decision-making processes regarding investments in financial assets. The research thesis proposes that implementing a BI system in an organization can support such decision-making processes. The study describes the structure and benefits of BI systems and presents a simplified project for the implementation of a BI system focused on financial asset investments. The research methods employed include a critical analysis of the literature and a case study.

## 2 Understanding the Significance and Functionality of Business Intelligence Systems

Managing an organization effectively requires the use of business intelligence (BI) systems, which plays a crucial role in strategic planning, monitoring economic processes, and building relationships with internal and external stakeholders. Initially, BI was defined as an integrated set of tools, technologies, and programmed products used to collect, interpret, analyze, and share data [1].

However, over time, it has evolved to become a factor connecting various components of decision support infrastructure and providing specific information to decision-makers [2]. Business intelligence is a term that refers to the theory of management, it is understood as an instrument used to support the organization in using and optimizing knowledge and making business decisions [3]. BI is characterized by the ability to provide information to interested and authorized persons at the desired time. BI helps in decision-making and improves data quality, operational efficiency, competitive advantage, and customer satisfaction [4]. Business intelligence is a system for transforming data into information (technological context), and information into knowledge (system context), which is used, e.g., to increase the value of an organization, enhance its competitiveness, and support business decision-making (management context).

Such computerized analytical platforms give users quick access to the necessary, unified, aggregated, and multi-criteria processed information. Thanks to such “data warehouses,” it is possible to use the built-in report or analysis templates and conduct independent analyses or reports tailored to the specific needs of the organization. As noted by Beer [5], the power of data lies mainly in the fact that they are used to predict and make decisions. The data themselves come to life and begin to have consequences when analyzed and integrated into specific structures. Very often, BI systems are “tailor-made” systems. BI gives managers the possibility of exploration, integration, aggregation, and multidimensional analysis of large datasets, which increases the efficiency of company management, ultimately, indirectly building the organization’s

value. BI also improves the transparency of information flow, allowing users to detect and prevent business anomalies and fraud. Thanks to BI, standardized reports can be generated for given areas of the organization's activity and key performance indicators can be calculated. On their basis, hypotheses are put forward, which are then verified by performing detailed information "cross-sections" using analytical tools (e.g., OLAP cube—Online Analytical Processing, data mining). Therefore, BI is the ability to plan, anticipate, solve problems, increase organizational knowledge, and provide information for the decision-making process, enabling effective action and supporting business goals.

The BI technological structure consists of at least the following elements [6]:

- ETL tools (extract, transform, and load) supporting the processes of data acquisition and storage, building or obtaining a "data warehouse,"
- Data mining—data extraction, data drill down,
- OLAP cube, including processing and comprehensive data analysis with a presentation, data mining tools to discover regularities, patterns, generalizations, and rules in data resources,
- Tools for reporting and queries (e.g., dashboards)—the report presentation with graphical and multimedia interfaces, thanks to which information is provided in a convenient, suggestive form to users.

The information in an OLAP cube is often presented in the form of a multidimensional cube, which includes various data processing methods as dimensions. These cubes, which can contain more than three dimensions, are often referred to as hypercubes. The cube is essentially a structured set of data, typically found in multidimensional spreadsheets, which can be used as a multi-scale model to optimize management. It is designed to allow users to analyze and manipulate data from different angles, using different dimensions. Common operations include roll up (i.e., generalizing data), roll down (i.e., refinement), slicing, projection (i.e., reducing the number of dimensions), cutting (i.e., combining selection with projection), sorting (i.e., creating rankings), and rotating (i.e., changing the perspective of viewing data). The multidimensionality of the cube depends on the user's needs and the complexity of the system. It is important to note that BI solutions should be scalable and the cube's architecture should be designed for expansion as needed. It is also crucial that the system is based on modern technologies.

Access to the information contained in the multidimensional cube is dependent on the user's authorization and responsibilities. These information rights should be defined within the system. Senior management will have greater access than lower-level employees, and there should be a thematic division related to job responsibilities.

To sum up, the use of a BI system can increase information benefits, including the following advantages:

- increasing the availability of information, especially during a pandemic, information is collected in one database, which facilitates access to it, especially when working remotely,

- increasing efficiency—data are centralized, structured, time-saving, and streamlining the decision-making process,
- greater support in decision-making compared with “regular” information systems,
- determining the causes of plan failures and eliminating these factors,
- streamlining systems monitoring the process execution and detecting the causes of irregularities during the implementation,
- carrying out transformations in the organization, i.e., introducing a new business model focused on change management, knowledge management, customer relationship management, etc. [7].

Knowledge, to be effectively used in the decision-making process, should be stored and created according to proven research methods, and it should also be updated on an ongoing basis. Today, solutions based on artificial intelligence, including fuzzy logic, intelligent agents, genetic algorithms, natural language processing, and CBR are gaining particular importance.

### 3 Review of Business Intelligence Literature

Several literature reviews on current BI aspects in enterprise accounting for an industry perspective have been published in the last decade [8–12]. Numerous literature reviews on the use of BI in various industries have been carried out, e.g., in tourism by Nyanga et al. [13]; in higher education by Jamiu et al. [14]; in controlling and management accounting by Peters et al. [15]. Nithya and Kiruthika [3] conducted a literature meta-analysis on BI in the banking sector. A proposal for a framework for the successful development of a BI system in the health sector, taking into account organizational, process, and strategic conditions was presented by Kitsios and Kapetaneas [16]. According to research conducted by Papadopoulos and Kanellis [17], the successful implementation of business intelligence (BI) in an organization relies on the expertise of information management managers. This includes everything from collecting data to delivering it to the appropriate personnel throughout the company’s organizational structure. Similarly, a study by Katarina et al. [18] found that BI tools rely heavily on information technologies like online analytical processing and data mining. Furthermore, an article by Bogdan and Emin [19] demonstrated how a BI system can be used to facilitate high-quality and timely decision-making in managing a bank’s assets and liabilities. In addition, Anuj et al. [20] highlighted the impact of statistical tools when implemented through an appropriate BI framework on fraud detection. While there is a significant amount of theoretical research on BI, there is also some applied research available. Table 1 provides an overview of some selected studies in the field of BI, along with their findings.

The presented research did not analyze the BI impact on aspects related to financial assets. Still, taking into account the various success factors in multiple industries presented in the literature, i.e., organizational, process, technological, and above all, managerial support [29], the authors recognized as reasonable to present an original

**Table 1** BI literature overview—conducted research and results

Author	Research area	Research results
Sujitparapitaya et al. [21]	A study based on 243 higher education institutions in the United States investigated the technological, organizational, and environmental factors that influence BI adoption in private and public higher education institutions	Institutions of higher education were less likely to adopt BI if they were private rather than public; and instead of being a deterrent, the perceived complexity of BI applications was positively related to BI adoption in academic institutions
Tuncay and Belgin [22]	Analysis of the satisfaction level with decision support systems (ERP) and BI class systems based on 25 companies from Turkey	Research on BI class systems showed that only 4% of enterprises declared their use, and awareness of the use of these systems is low
Barua et al. [23]	Research on 150 Fortune companies	The authors estimated the impact of improving information quality provided in organizations on enterprises' financial indicators of profitability, innovation, and operational efficiency
Olszak [24]	The study aimed to assess BI use in twenty deliberately selected organizations and determine the factors that allow companies to achieve high competence in BI through in-depth interviews	Among the twenty organizations surveyed, two fell into the category of the highest BI maturity level. These were a telecommunications company and a marketing agency. Their BI competencies are focused on competing and achieving business benefits, such as acquiring new customers
Al-Shubirii [25]	The research concerned 50 companies listed on the Stock Exchange in Amman in 2006–2010 and examined the impact of BI on the company's profitability	The study defined four areas where BI influenced ROI, customer satisfaction, innovation and learning ability, and intellectual capital. The impact of BI on profitability measured by ROE has been proven
Obidat et al. [26]	The research was based on 462 survey answer sheets among administrative employees, aimed at examining the impact of BI on supply chain agility within the Jordanian manufacturing sector	Results revealed that the three dimensions of BI have statistically significant positive direct effects on supply chain agility. In addition the results revealed that BI cultural competence has statistically significant positive direct effects on BI technical and managerial competencies
Kaur [27]	This study examined how BI implementation affects the agile efficiency of the supply chain with the logistics industry's supply chain responsiveness—the survey of 50 respondents	The study found that BI competence has a significant positive impact on the response to the supply chain, has a significant positive impact on the supply chain's agile performance, and has a significant positive impact on agile performance

(continued)

**Table 1** (continued)

Author	Research area	Research results
Rouhani et al. [28]	A survey of 228 companies from various industries from Middle Eastern countries aimed at presenting the relationship between BI capabilities, decision support benefits, and organizational benefits in the decision-making environment	The results confirmed the existence of a significant relationship between BI, decision support benefits, and organizational benefits in the decision-making environment, with 15 out of 16 hypotheses confirmed
Olszak and Ziembka [6]	BI systems success in small and medium-sized enterprises was examined. In this study, authors used an analysis of static data and literature, in-depth interviews as well as critical thinking and inductive inference	The most important factors for BI systems success for managers were an appropriate budget, well-defined business processes, and user expectations, as well as BI system integration with other systems

Source own study based on [6, 21–28]

BI system in a hypothetical enterprise and indicate the benefits of implementing BI for investments in financial assets.

#### 4 Traditional Information Systems on Investments in Financial Assets

Financial asset data have traditionally been obtained from accounting and reporting systems, which collect detailed information on an ongoing basis. Financial statements provide information on the value of financial assets (balance sheet), profits or losses from their sale (income statement), and additional details such as write-downs and changes in asset value by classification groups. Furthermore, financial asset risks are reported separately and analysts collect and analyze information independently. Information is partly published in the business model, though it is a report not audited by a statutory auditor.

The characteristics of this information system on investments in financial assets are:

- a large amount of data that is scattered or inaccessible to all users,
- the need to duplicate and transfer data to make them available to interested organizational units,
- lack of a uniform data format, information prepared according to incomparable criteria and in different styles, which makes data analysis difficult,
- lack of a separate analysis and reporting system that would generate data useful to interested organizational units,
- systematic data update.

Still, traditional systems do not provide key data necessary for objective assessment and analysis by managers. Moreover, rescaling and presenting data in a different context is laborious. External stakeholders only have access to financial statements, which do not allow for a detailed assessment of an organization's potential to create value. This causes a growing information disproportion between those with access to detailed but incomplete information and those making investment decisions based only on publicly available financial statements. This can lead to a non-objective market valuation of the organization, causing deep price fluctuations in the capital markets and misallocation of investment funds at the enterprise and capital market levels.

## **5 Business Intelligence for Investments in Financial Assets—The Case of an IT Industry Organization**

### **5.1 Preliminary Information**

The chosen organization previously used the traditional method of duplicating documents to create and transfer information on investments in financial assets, which were heterogeneous, in different formats, and their preparation was laborious. To address this issue, the organization decided to implement a BI system supplying information to the entire organization. However, only one branch of the organization and the information area regarding investments in financial assets will be subject to the study. The BIFA system aims to provide internal users with information on the current state of business processes related to financial assets, including acquisition, changes, and risk. The proposal of the business intelligence architecture for investments in financial assets (BIFA) aims at meeting the information needs of a large group of internal users of the organization regarding the current state of business processes related to the acquisition, changes, and risk of financial assets, informing about activities that are not in line with the schedules, etc. The modified BIFA system architecture consists of adapting the traditional system toward more streamlined business processes related to investments in financial assets. Still, the BIFA system implementation requires appropriate infrastructure, resources, and approach to design and use. Therefore, the following factors are essential: technological (infrastructure efficiency, data integrity, selection of IT tools), business (business justification, management support), and organizational (interactive approach to development, selection of management methods and techniques).

An important business aspect of building BIFA is to define the internal and external stakeholders of the company for whom the information is created, the company's business goals, and the possibilities of achieving them, and to indicate the potential benefits of its use.

The internal stakeholders interested in implementing BIFA in the examined branch of the organization are the following:

- the accounting department, including the chief accountant and an employee assigned to record investments in financial assets,
- risk management department, including an employee managing market risk, an employee managing operational and credit risk, and the head of the department,
- the financial and non-financial reporting department, including the person responsible for the valuation of financial assets, the person responsible for reporting on the business model, and the head of the department,
- the controlling department, the person responsible for analysis preparation,
- the department of investments in financial assets, including persons responsible for investments in financial assets and the head of the department,
- enterprise management.

External stakeholders include other branches of the organization and head office, tax offices, banks, customers, and other users of financial and non-financial statements.

Efficient information acquisition is a multistage process. Before analysis determining the sources and ways of obtaining information imported into the data warehouse should be the analysis and design of critical functions (including assigned activities), responsibility for them, and data flow streams.

This is a critical stage of implementing the BIFA system, particularly in the analyzed entity where users of individual systems are required to model, verify, and code knowledge because they know the specifics of their business systems. This process should be followed by parameterization, including the introduction of knowledge necessary for the proper system operation (collection of data circulating in the integrated management system and other systems) in the data warehouse and user training. The data warehouse preparation is the basis for the operating mechanisms of the modern BIFA system. In the analyzed entity, useful data will be downloaded from source systems and supplemented with data identified during the creation of the necessary reports. Supplying the warehouse (ETL, extract, transform, load) with information on investment assets consists of extracting data from sources related to management, and entering and transforming information on financial assets and their risk. Then, the aggregated data are subjected to appropriate transformations to prepare them for target reports. Collecting data in a data warehouse brings advantages as data on financial investments are saved from the systems at specific time intervals, and previously copied, without deleting them, become historical data, which allows stakeholders to assess and analyze the dynamics of changes, e.g., changes in impairment losses on financial assets in particular periods.

## ***5.2 Original Business Intelligence System Employed in the Organization***

A basic OLAP cube was distinguished in the BI system and divided into seven dimensions (objective cognitive limitations do not allow for a graphical presentation of the main cube). These dimensions include:

1. Information on financial assets.
2. Information on the risk of financial assets.
3. Geographical area of the enterprise.
4. Time period.
5. Funding sources.
6. External reporting.
7. Internal reporting.

The first dimension of information on investment in assets creates the following:

- information on the classification groups of financial assets and the structure of investments,
- information on the value (valuation) of financial assets,
- information on write-downs on financial assets,
- information on the results of investments in financial assets,
- information on the effects of changing valuation methods of investments in financial assets.

The second dimension—information on risk related to financial assets—consists of the following:

- information on market risk related to investments in financial assets (share price risk, currency risk, interest rate risk),
- information on credit risk related to investments in financial assets,
- information on operational risk related to investments in financial assets,
- information on risk hedging instruments related to investments in financial assets,
- information on open positions exposed to the risk of investment in financial assets,
- information on risk management of investing in financial assets,
- information on the effects on the financial result and other effects of risk management of investments in financial assets.

The third dimension of information in the OLAP cube is the geographical dimension, where we can distinguish information divided into country, region, province, and city because the organization has branches in various locations in Poland and abroad. In addition to normal operating activities, each branch deals with investments in financial assets and manages risk by closing or not closing open positions exposed to risk. Funds for investments come from the head office, which evaluates the performance of individual branches in this area.

The fourth dimension is time. Information is created and processed for the following periods: year, quarter, month, and day.

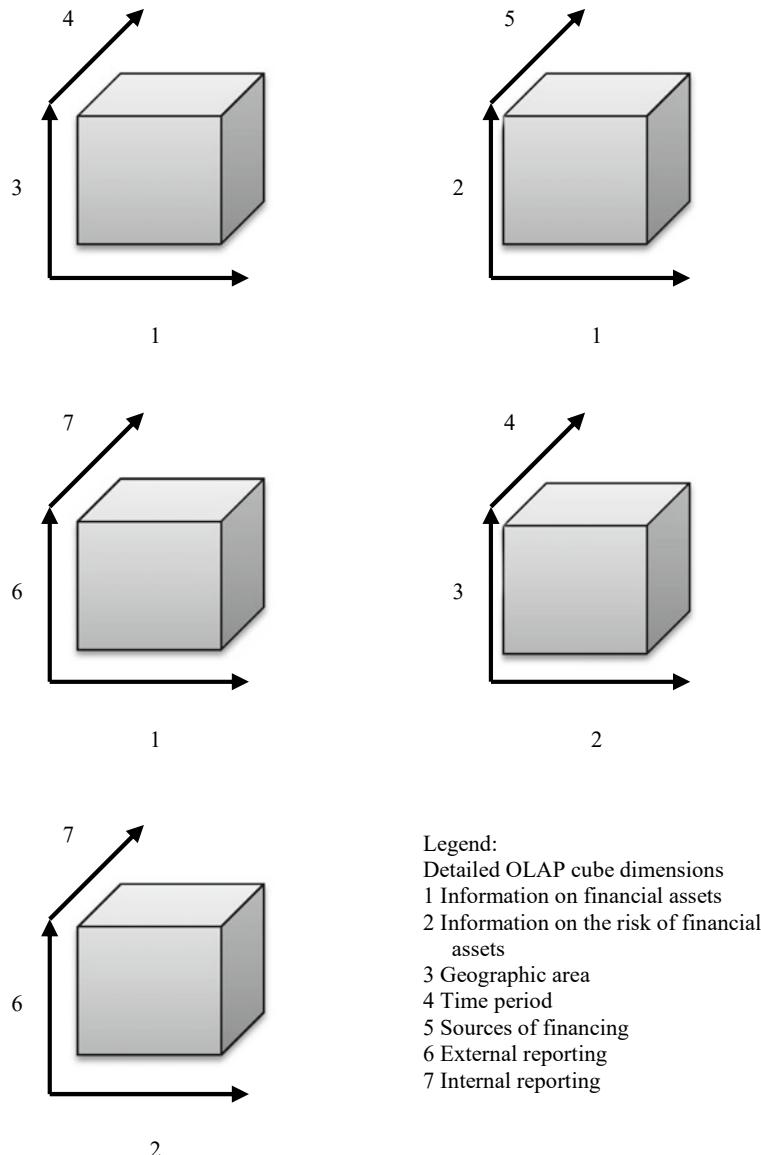
The fifth dimension of information is the sources of financing financial assets. At the very least, the following are distinguished:

- loans granted by other branches of the organization,
- bank credits and loans,
- a fund pool to finance investments in a given branch,
- a fund pool at the disposal of the branch manager.

The sixth dimension is reports intended for external publication, and the seventh dimension is reports created for internal needs, i.e., confidential, addressed to selected managers and line managers of the organization. Each branch generates financial and non-financial statements, while the head office generates consolidated reports. Concerning the risk information, it will be essential to describe the business model in the non-financial report, which includes information on risk factors and its effects [30].

The following information tables have been designed in the organization by combining selected dimensions of an OLAP cube, which can be called “detailed OLAP cubes,” and are presented in Fig. 1:

- Dimensions 1, 3, and 4, i.e., information on investments in financial assets broken down by geographical area and time. It is possible to roll down or up the cube, i.e., information about the selected branch or group of branches and the selected period. Based on the introduced criteria, a whole range of important information can be selected, e.g., the number of write-downs on financial assets in given periods and selected geographical areas,
- Dimensions 1, 2, and 5, i.e., information on investments in financial assets and information on the risk of those assets (risk can be assigned to groups of assets) broken down by financing sources of investments in financial assets. Thanks to this combination, it is possible, for example, to assess the centers of responsibility for investments and to assess the performance of the branch manager who is at the disposal of their own funds (if an investment in financial assets was made),
- Dimensions 1, 6, and 7, i.e., information on investments in financial assets broken down into external and internal (confidential) reports. This makes it possible to divide and dedicate information to the appropriate organizational units and authorized persons,
- Dimensions 2, 3, and 4, i.e., information on individual types of investment risk in financial assets and hedging instruments, risk management, and its effects broken down by geographical area and time. It is possible to roll down and up the cube, i.e., information about the selected branch or group of branches and the selected period,
- Dimensions 2, 6, and 7, i.e., information on individual types of investment risk in financial assets and hedging instruments, risk management, and its effects broken down into external and internal (confidential) reports. This makes it possible to divide and dedicate information to the appropriate organizational units and authorized persons.



**Fig. 1** Detailed OLAP cubes

Thanks to such a structure of entity's detailed OLAP cubes, it will be possible to quickly and automatically create formally unified, various reports, using, e.g., data mining functionality, thanks to which stakeholders will receive information on the share of individual branches in financial risk broken down into specific periods,

along with the possibility of its financing classified into internal, confidential, and published information.

Information from individual dimensions goes to authorized stakeholders. In a hierarchy, data can be organized into lower and higher levels of detail according to the information needs. OLAP data come from historical data, but they also contain forecasts and are compiled into structures, thanks to which their advanced analysis is possible. OLAP data are also organized hierarchically and stored in cubes instead of tables. It is an advanced technology that uses multidimensional structures that provide quick access to such data and their subsequent analysis. OLAP databases in BI have been designed so that data transfer is as fast as possible. The OLAP server calculates summary values, sending fewer data to the server while creating or changing reports. This method allows users to obtain and use much more source data than if the data were organized in a traditional database. OLAP databases contain two basic types of data: measures, which are numerical data, quantities, and averages used to make business decisions, and dimensions, which are categories used to organize these measures. OLAP databases make it easy to manage data based on multiple levels of detail.

The effect of BIAF adaptation in the organization can yield multidimensional benefits that can be divided into:

1. Market benefits:

- improving the quality of information for external customers.

2. Benefits of internal (organizational) processes:

- positive impact on decision-making, thanks to quick access to processed and aggregated information in spreadsheets, also in the form of charts and presentations,
- knowledge transfer between teams and individual branches of the organization,
- better risk management of financial assets (decision-making based on ratios, dashboards),
- optimizing the processing of information on investments in financial assets.

3. Growth and development benefits:

- greater operational efficiency in investments in financial assets, thanks to reliable information support,
- creating the value of the organization,
- probable increase in the value of the investment in the organization,
- probable positive impact on the organization's bottom line.

The Covid pandemic led to a significant decrease in organizations' financial results, and even to their collapse. Digital solutions were important in responding to the impacts of the pandemic. To prepare for possible future threats (e.g., cyber-attacks, armed conflicts), organizations should already be implementing BI systems whenever possible and use the capabilities of these systems by expanding critical

information, e.g., the information presented about financial assets. This may support managers in making the right decisions, especially in times of a pandemic. Thanks to the creation of the presented BIAF system, it will be possible to secure the financial result by quickly responding to the emerging risk of financial assets. Automation and standardization of the information system can contribute to quick access to large collections of transparent [31], standardized, and strategic information.

## 6 Conclusions

The main purpose of BI systems is faster and more comprehensive assistance in making various decisions, thus creating a competitive advantage and the organization's value. In addition, thanks to statistical methods, BI systems can capture market developments or trends. Thanks to timely and full access to standardized operations, managers can make more and more effective decisions, also in the area of investing in financial assets.

Therefore, business intelligence systems have a multidimensional impact on the organization. Organizations supplied with information, including information on financial assets operate more effectively because individual organizational units directly and indirectly related to these aspects are better managed and make more accurate decisions. BI systems:

- help to make decisions based on quickly obtained and multidimensional information and analyses, thus reducing uncertainty and risk,
- help to quickly extract relevant information from the system and present it in a convincing, standardized form,
- affect the management mode of the entire organization, contributing to the achievement of better results and creating the organization's value.

The weakness of investing in BI is certainly the cost and the human factor. Employees must be willing to use this information, and additional training and proper staffing may be required.

The study presents a selected, small area of BI related to investments in financial assets because the description of the entire system is extremely extensive. Nevertheless, the authors want to continue research in the field of BI in other areas of the organization's activity.

## References

1. Reinschmidt J, Francoise A (2020) Business intelligence certification guide. IBM, International Technical Support Organization. San Jose CA

2. Stylos N, Zwiegelaar J (2019) Big data as a game changer: how does it shape business intelligence within a tourism and hospitality industry context? In: Big data and innovation in tourism, travel, and hospitality: managerial approaches, techniques, and applications, pp 163–181
3. Nithya N, Kiruthika R (2021) Impact of Business Intelligence Adoption on performance of banks: a conceptual framework. *J Ambient Intell Humaniz Comput* 12:3139–3150
4. Ritacco M, Carver A (2007) The business value of business intelligence. A framework for measuring the benefits of business intelligence. *Bus Objects* 1–24
5. Beer D (2018) Envisioning the power of data analytics. *Inf Commun Soc* 21(3):465–479
6. Olszak CM, Ziembka E (2012) Critical success factors for implementing business intelligence systems in small and medium enterprises on the example of upper Silesia, Poland. *Interdiscip J Inf Knowl Manag* 7:131
7. Goodhue DL, Wixom BH, Watson HJ (2002) Realizing business benefits through CRM: hitting the right target in the right way. *MIS Q Exec* 1:79–94
8. Eggert M, Alberts J (2020) Frontiers of business intelligence and analytics 3.0: a taxonomy-based literature review and research agenda. *Bus Res* 13(2):685–739
9. Ain N, Vaia G, DeLone WH, Waheed M (2019) Two decades of research on business intelligence system adoption, utilization and success. A systematic literature review. *Decis Support Syst* 125
10. Jourdan Z, Rainer RK, Marshall TE (2008) Business intelligence: an analysis of the literature. *Inf Syst Manag* 25(2):121–213
11. Rashid A, Khurshid MM (2022) A Descriptive literature review and classification of business intelligence and big data research. In: Intelligent computing. Proceedings of the 2022 computing conference, vol 1. Springer International Publishing, Cham, pp 865–879
12. Purnomo A, Firdaus M, Sutiksono DU, Putra RS, Hasanah U (2021) Mapping of business intelligence research themes: four decade review. In: 2021 IEEE international conference on communication, networks and satellite (COMNETSAT). IEEE, pp 32–37
13. Nyanga C, Pansiri J, Chatibura D (2020) Enhancing competitiveness in the tourism industry through the use of business intelligence: a literature review. *J Tourism Futures* 6(2):139–151
14. Jamiu SM, Abdullah NS, Miskon S, Ali NM (2020) Data governance support for business intelligence in higher education: a systematic literature review. *Emerg Trends Intell Comput Inform Data Sci Intell Inf Syst Smart Comput* 4:35–44
15. Peters MD, Wieder B, Sutton SG, Wakefield J (2016) Systemy Business Intelligence wykorzystują możliwości pomiaru wydajności: implikacje dla zwiększenia przewagi konkurencyjnej. *Int J Account Inf Syst* 21:1–17
16. Kitsios F, Kapetaneas N (2022) Digital transformation in healthcare 4.0: critical factors for business intelligence systems. *Information* 13(5):247
17. Papadopoulos T, Kanellis P (2010) A path to the successful implementation of business intelligence: an example from the Hellenic Banking Sector. *OR Insight* 23(1):15–26
18. Katarina C, Mirjana PB, Gordana R (2008) Business intelligence and business process management in banking operations. In: International conference on information technology interfaces, Cavtat
19. Bogdan U, Emin D (2011) Application of business intelligence in the banking industry. *Manag Inf Syst* 6(4):23–30
20. Anuj S, Parbin Kumar P (2012) Review of financial accounting fraud detection based on data mining techniques. *Int J Comput Appl* 39(1):37–47
21. Sujitparapitaya S, Shirani A, Roldan M (2012) Business intelligence adoption in academic administration: an empirical investigation. *Issues Inf Syst* 13(2):112–122
22. Tuncay EG, Belgin O (2023) Effects of business intelligence techniques of enterprise productivity. [www.academia.edu/2700385](http://www.academia.edu/2700385). Last accessed 02 May 2023
23. Barua A, Mani D, Mukherjee R (2023) Measuring the business impacts of effective data. University of Texas at Austin study. [www.sybase.com/files/White\\_Papers](http://www.sybase.com/files/White_Papers). Last accessed 02 May 2023
24. Olszak CM (2013) Organizacja oparta na Business Intelligence-wybrane wyniki badań empirycznych. *Studia Ekonomiczne* 136:231–244

25. Al-Shubirii FN (2012) Measuring the impact of business intelligence on performance an empirical case study. *Pol J Manage Stud* 6
26. Obidat A, Alziyat Z, Alabaddi Z (2023) Assessing the effect of business intelligence on supply chain agility. A perspective from the Jordanian manufacturing sector. *Uncertain Supply Chain Manage* 11(1):61–70
27. Kaur K (2021) Business intelligence on supply chain responsiveness and agile performance: empirical evidence from Malaysian logistics industry. *Int J Supply Chain Manage* 6(2):31–63
28. Rouhani S, Ashrafi A, Zare Ravasan A, Afshari S (2016) The impact model of business intelligence on decision support and organizational benefits. *J Enterp Inf Manage* 29(1):19–50
29. Adamala S, Cidrin L (2011) Key success factors in business intelligence. *J Intell Stud Bus* 1(1):107–127
30. Szewieczek A, Dratwińska-Kania B, Ferens A (2021) Business model disclosure in the reporting of public companies—an empirical study. *Sustainability* 13(18)
31. Dratwińska-Kania B, Ferens A, Kania P (2023) Transparent reporting on financial assets as a determinant of a company's value—a Stakeholder's perspective during the SARS-CoV-2 pandemic and beyond. *Sustainability* 15(3)

# Feature Selection Techniques to Enhance Prediction of Clinical Appointment No-Shows Using Neural Network



Jeffin Joseph, S. Senith, A. Alfred Kirubaraj, and S. R. Jino Ramson

**Abstract** The issue of no-shows is a significant concern, as it results in many patients missing their appointments at outpatient clinics worldwide, often without any prior cancellation. This leads to inefficiencies in terms of idle resources and wasted capacity. Hence prediction models are needed to anticipate whether a patient will attend their scheduled appointment. The effectiveness of a predictive model in predicting clinical appointment no-shows is heavily influenced by the features used in the model. To address this issue, the present study compares various feature selection techniques in order to enhance the accuracy of prediction. Univariate Selection, Recursive Feature Elimination, Random Forest Classifier, and Reciprocal Ranking are the feature selection techniques utilized in the current study. These techniques are applied prior to building a neural network model using a Multilayer Perceptron to predict clinical appointment no-shows. The study employed the scikit-learn library in python for model implementation, and the performance of each model is evaluated using performance measures, including Accuracy, Specificity, Sensitivity, Precision, F-measure, Matthews Correlation, Log Loss, and Area under the Curve. Feature selection methods demonstrated excellent performance by reducing the number of variables while maintaining the predictive accuracy across all models. Consistently, the most critical features across all models were the Patient Trust, Appointment Type, Reminder Message, Lead Time, and Missed Appointment History.

**Keywords** Hospital management · Appointment no-shows · Neural network · Predictive analytics · Feature selection methods

---

J. Joseph (✉) · S. Senith · A. Alfred Kirubaraj  
Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India  
e-mail: [jeffinjoseph@hotmail.com](mailto:jeffinjoseph@hotmail.com)

S. R. J. Ramson  
GlobalFoundries US LL2, Vermont, USA

## 1 Introduction

Feature selection facilitates in identifying the crucial features in predictive analytics problems. The features that are chosen typically affect how well a prediction model performs. The computational complexity of the model is also increased due to unnecessary and duplicate features [1, 2]. Filter, Wrapper, and Embedded approaches are the broad categories in feature selection. Wrapper methods use a machine learning algorithm as a black box evaluator to identify the optimal subsets of features, but they are dependent on the classifier. Filter methods, on the other hand, rank features based on their intrinsic properties, independent of any machine learning algorithm. Embedded methods combine quantitative and statistical criteria by first applying a filter to select some features, and then using a machine learning algorithm to choose the best performing subset [3, 4]. The current paper addresses the issue of predicting the clinical appointment no-shows using neural network particularly a Multilayer Perceptron. No-shows are critical problem in healthcare sector worldwide as they can have a negative impact on patients' health, resulting in treatment discontinuity and inefficient utilization of hospital resources. Prediction of no-shows in advance will allow for the development and implementation of intervention strategies more effectively to reduce the uncertainty of no-shows. Any clinic that permits patients to make appointments in advance runs the risk of the patient missing the appointment or canceling it without giving enough notice. Because of this, a lot of hospitals rely on the pricey strategy of sending reminders, which is commonly combined with overbooking, the act of scheduling numerous patients in a single appointment time. The massive gathering of data in electronic health records of hospitals, along with advancements in machine learning, allowed the researchers to conduct more specialized clinical no-show prediction. Despite these developments, there are still unresolved issues with the prediction process. One of the main issues is in selecting the most relevant features for developing the prediction model. The features employed in the current study for prediction are Patient Age, Patient Gender, Appointment Type, Missed Appointment History, Patient Trust, Appointment Weekday, Lead time, Reminder Message, Hypertension, Scholarship, Diabetics, Alcohol Addiction, and Handicap. Feature selection methods are implemented before applying neural network approach for the predicting the no-shows. Univariate Selection, Recursive Feature Elimination, Random Forest Classifier, and Reciprocal Ranking are the feature selection methods used in the study. The Multilayer Perceptron was chosen for prediction, in this study due to its superior performance in comparison to other machine learning approaches for classification problems [5]. In the current study, it was observed that feature selection methods demonstrated excellent performance by reducing the number of variables while maintaining the predictive accuracy across all models. Consistently, the most critical features across all models were the Patient Trust, Appointment Type, Reminder Message, Lead Time, and Missed Appointment History.

The rest of this paper is organized as follows. Section 2 gives a brief review of the related studies. Section 3 explains the development of model. The analysis of performance measures and discussion are given in Sect. 4. Conclusions, and future work is drawn in Sect. 5.

## 2 Related Studies

Inclusion of irrelevant features during model training reduces the predictive model's overall accuracy while increasing its complexity and bias. Additionally, its ability to generalize to new data is reduced [6]. The adage "Sometimes less is better" applies to the learning model as well. Thus, one of the crucial steps in creating a prediction model is feature selection. Feature selection eliminates irrelevant and noisy features and keeps those with high relevance to the target variable. This reduces computational time, improves algorithm performance, and creates better general models by avoiding overfitting. However, optimal feature subsets may not be unique due to inter-feature correlation [7].

Univariate Selection is a filter method that orders features based on their intrinsic properties such as variance, consistency, distance, information, correlation, and is advantageous for performing feature selection before classification without the risk of learning algorithm bias interacting with feature selection algorithm bias [8]. The Recursive Feature Elimination method searches for a subset of features by eliminating features until the desired number remains. It is effective at selecting relevant features for predicting the target variable and is easy to use with two important configuration options: the number of features to select and the algorithm used to choose features [9]. Random Forest Classifier is an embedded method for feature selection. Embedded methods combine statistical and measurable criteria like filters and then use a machine learning algorithm to pick the best subset for classification. However, these methods are dependent on the classifier since feature selection is done during the learning phase [10, 11].

Researchers have developed ensemble feature selection methods in recent years, which produce more reliable results than using a single algorithm. The goal of ensemble method is to integrate many feature selection techniques, considering each one's advantages, to provide the best possible subset [12, 13]. Using experiments on eight environmental datasets, a study examined various feature selection methodologies and determined that wrapper methodologies perform better than filter methodologies, while embedding methodologies work about as well. The study also employed various ensemble approaches, and Reciprocal Ranking fared better than other models [14].

In another paper, two feature selection techniques were proposed to select features. The first technique used Information Gain and Forward Selection, while the second technique used Recursive Feature Elimination with SVM. Both techniques were then applied with Rough Set Theory to improve the performance [15]. In another study, four different feature selection methods including Principal Component Analysis

were applied to evaluate four heart disease datasets and develop different feature sets for predicting heart conditions. These feature sets were used to create models using various classification algorithms in order to improve the accuracy of heart condition predictions [16]. Another study by Yang et al. compared different feature selection algorithms for cardiovascular disease prediction [17].

Many studies have found that deep learning techniques, such as Multilayer Perceptron, are the most effective predictive models for predicting appointment no-shows. Recent research has investigated the performance of Deep Neural Network (DNN), AdaBoost, and Naive Bayes (NB) algorithms in predicting no-shows. The study found that the DNN algorithm outperformed both AdaBoost and NB in accurately predicting no-shows. The DNN algorithm is particularly effective in modeling complex relationships between input data and output predictions, which may explain its superior performance compared to the other two algorithms [18]. Another study that was conducted out in an academic pediatric teaching hospital employed Logistic Regression and neural network approaches to predict appointment no-shows and discovered that the patients' prior no-show history is the best predictor [19].

An end-to-end deep learning model based on sparse stacking denoising autoencoders (SSDAE) was developed by Dashtban and Li in a 2019 study to predict appointment no-shows [20]. Another study used Logistic Regression, Artificial Neural Network, and Naive Bayes Classifier models to predict missed appointments and to identify crucial variables for prediction and discovered lead time, patient prior missed appointments, ownership of mobile, tobacco and number of days since last appointment are crucial variables [21]. There are dozens of other studies which designed machine learning models using Logistic Regression, Support Vector Machines, Random Forests, AdaBoost, K-Nearest Neighbor (KNN), Boosting, Decision Tree (DT), Random Forest (RF), Bagging and Stochastic Gradient Descent, for prediction of no-shows [22–24]. The performances of the models are analyzed using different performance measures such as Geometric Mean,  $F_1$  score, and AUC curve.

### 3 Model Development

The data used for the study consists of 110466 appointments of patients in a hospital in Brazil. The original dataset consisted of variables like Patient Id, Appointment Id, Patient Gender, Scheduled Date, Appointment Date, Age, Scholarship, Hypertension, Diabetes, Alcohol Addiction, Handicap, Reminder Message, and No-show. The raw dataset is cleaned, and some new variables are extracted for better prediction of the appointment no-show. Appointment Type, Missed Appointment History, Patient Trust, Appointment Weekday, Lead Time are the new variables extracted from the dataset. The process of extracting new variables from existing variables is called feature engineering [25]. The continuous variables are converted to categorical variables for better prediction. The continuous variable Age is converted to categorical variable Patient Age with categories Senior, Adult, Young Adult, Adolescent, and Child. Appointment Type consists of Fresh and Repeated which is derived from the

Patient Id and Appointment Id data. Missed Appointment History indicates whether patients have the history of missed appointments. Patient Trust is a derived variable consisting of four categories. The categories are based on the patient's history of missed appointments. A value of 0 indicates no missed appointments and high trust, while a value of 1 indicates 2 or 3 missed appointments. A value of 3 corresponds to 4–10 missed appointments, and a value of 4 corresponds to a history of more than 10 missed appointments and has low trust. Week Day is a variable derived from the Appointment Day data which consists of categories Monday to Saturday. Lead Time is another derived variable calculated by taking the difference between Appointment Date and Scheduled Date. The Lead Time variable consists of categories Same day, Less than two days, Less than a week, Less than a month, and More than a month. Reminder Message consists of categories Yes and No, which shows whether the reminder message is send. Scholarship, Diabetics, Hypertension, and Alcohol Addiction also consists of categories Yes and No. The Variable Handicap consists of categories 0–4 which describes the number of disabilities of patients.

Categorical variables Patient Age, Patient Gender, Appointment Type, Missed Appointment History, Patient Trust, Appointment Weekday, Lead Time, Reminder Message, Hypertension, Scholarship, Diabetes, Alcohol Addiction, and Handicap are the variables used for prediction. A detailed analysis of the categorical variables used in the study with appointment no-shows is given in Table 1.

After preprocessing the data five models are developed Model 1. Using all variables, Model 2. Using variables obtained from feature selection using Univariate Selection, Model 3. Using features from Recursive Feature Elimination, Model 4. Using features obtained from feature importance using Random Forest and Model 5. Using features from Reciprocal Ranking, where it aggregates the ranks calculated using feature importance scores obtained through Extra Trees and XGBoost. Then the data is split into training set and test set in the ratio 80:20. The training set is trained using the Multilayer Perceptron Classifier in the scikit-learn library and is tested with the test set. The performance is analyzed using evaluators like Accuracy, Precision, Specificity, Sensitivity, F-measure, Matthews Correlation, Log Loss, and Receiver Operating Characteristic Curve [26, 27]. The detailed framework of the prediction models is given in Fig. 1.

## 4 Analysis of Performance Measures

The performance measures are evaluated for all the models. The confusion matrix is evaluated followed by analyzing the performance measures such as Accuracy, Precision, Specificity, Sensitivity, F-measure, MCC, and Log Loss. Accuracy is a measure of how well a classifier distinguishes between positive and negative instances. It is the proportion of accurately predicted shows to the total number of instances.

Accuracy measures the overall correctness of the classifier's predictions, considering both positive and negative instances.

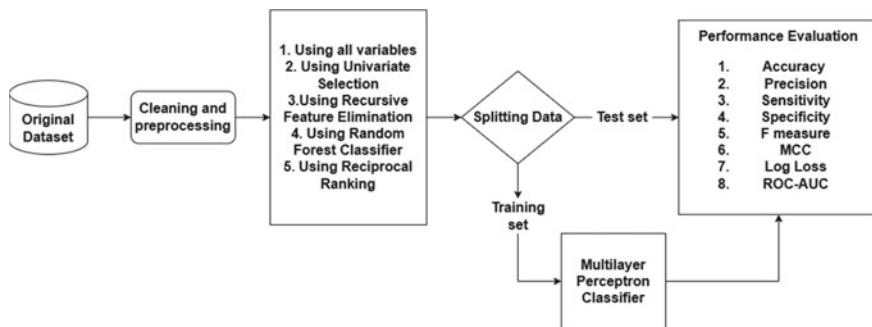
**Table 1** Descriptive analysis of all the variables

Variables		Total	Show	No-show
Patient age	Senior	25,284	21,026	4258
	Adult	52,053	41,451	10,602
	Young adult	6849	5124	1725
	Adolescent	6342	4652	1690
	Child	19,938	15,903	4035
Patient gender	Female	71,795	57,208	14,587
	Male	38,671	30,948	7723
Appointment type	Fresh	50,090	12,191	62,281
	Repeated	38,066	10,119	48,185
Missed appointment History	No	76,791		76,791
	Yes	11,365	22,310	33,675
Patient trust	4	76,791		76,791
	3	10,506	20,881	31,387
	2	847	1397	2244
	1	12	32	44
Appointment week day	Monday	18,019	4688	22,707
	Tuesday	20,470	5150	25,620
	Wednesday	20,747	5089	25,836
	Thursday	13,909	3337	17,246
	Friday	14,981	4037	19,018
	Saturday	30	9	39
Lead time	Same day	36,737	1791	38,528
	< Two days	9222	2712	11,934
	< a Week	15,187	5056	20,243
	< a Month	20,519	9544	30,063
	> Month	6491	3207	9698
Reminder Message	No	62,474	12,530	71,795
	Yes	25,682	9780	38,671
Scholarship	No	79,874	19,734	99,608
	Yes	8282	2576	10,858
Hypertension	No	70,133	18,541	88,674
	Yes	18,023	3769	21,792
Diabetics	No	81,646	20,880	102,526
	Yes	6510	1430	7940
Alcohol addiction	No	85,473	21,633	107,106
	Yes	2683	677	3360
Handicap	0	86,322	21,903	108,225

(continued)

**Table 1** (continued)

Variables		Total	Show	No-show
	1	1676	366	2042
	2	146	37	183
	3	10	3	13
	4	2	1	3

**Fig. 1** Framework of the study

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (1)$$

Precision focuses on the quality of the positive predictions, measuring the proportion of correctly predicted positive instances out of the total predicted positive instances.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (2)$$

Specificity (True Negative Rate) evaluates the classifier's ability to correctly identify the negative instances (shows), out of the total actual negative instances.

$$\text{Specificity} = \text{TN} / (\text{TN} + \text{FP}) \quad (3)$$

Sensitivity (Recall or True Positive Rate) measures the classifier's ability to correctly identify the positive instances (no-shows), out of the total actual positive instances.

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{FN}) \quad (4)$$

F-measure combines precision and sensitivity into a single metric, providing a balanced measure of the classifier's performance.

$$\text{F-measure} = 2 * ((\text{Precision} * \text{Sensitivity}) / (\text{Precision} + \text{Sensitivity})) \quad (5)$$

Matthews Correlation Coefficient (MCC) considers all four elements of the confusion matrix, providing a measure of the quality of binary classifications.

$$\text{MCC} = \frac{(TP * TN - FP * FN)}{\sqrt{((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))}} \quad (6)$$

Log Loss (Cross-Entropy Loss) is a loss function used to measure the performance of a classifier by comparing predicted probabilities with true class labels.

$$\text{Log Loss} = -(1/N) * \sum(y * \log(p) + (1 - y) * \log(1 - p)) \quad (7)$$

AUC evaluates the performance of a binary classifier based on its Receiver Operating Characteristic (ROC) curve, specifically for the prediction of no-shows. AUC is calculated by computing the integral of the ROC curve.

The initial model (Model 1) was evaluated using various performance measures. The results obtained are as follows: Accuracy–0.93, Precision–0.81, Specificity–0.95, Recall–0.87, F-measure–0.84, Matthews Correlation Coefficient (MCC)–0.80, Log Loss–2.45, and Area Under the Curve (AUC)–0.98. In the next step, Univariate Feature Selection was applied to choose 8 variables for prediction in Model 2. The selected variables used for prediction in Model 2 are Reminder Message, Lead Time, Missed Appointment History, Patient Trust, Appointment Type, Patient Age, Scholarship, and Hypertension. The performance measures obtained for Model 2 are as follows: Accuracy–0.93, Precision–0.81, Specificity–0.95, Recall–0.86, F-measure–0.84, MCC–0.80, Log Loss–2.42, and AUC–0.98. The performance measures for all the models are given in Table 2.

Model 3 utilized Recursive Feature Elimination to select following 8 variables for prediction: Appointment Type, Missed Appointment History, Patient Trust, Lead Time, Reminder Message, Hypertension, Scholarship, and Alcohol Addiction. The performance measures obtained for Model 3 are as follows: Accuracy–0.93, Precision–0.81, Specificity–0.95, Recall–0.87, F-measure–0.84, MCC–0.80, Log Loss–2.41, and AUC–0.98. Feature importance using Random Forest was employed to

**Table 2** Performance measures for the models

Performance Measures	Model 1	Model 2	Model 3	Model 4	Model 5
Accuracy	0.93	0.93	0.93	0.93	0.93
Precision	0.81	0.81	0.81	0.80	0.83
Specificity	0.95	0.95	0.95	0.94	0.96
Recall	0.87	0.86	0.87	0.91	0.85
F-measure	0.84	0.84	0.84	0.84	0.84
MCC	0.80	0.80	0.80	0.81	0.80
Log Loss	2.45	2.42	2.41	2.40	2.41
AUC	0.98	0.98	0.98	0.98	0.98

select 8 variables for prediction in Model 4. The variables used for prediction in Model 4, utilizing the Random Forest Classifier, are Patient Trust, Missed Appointment History, Appointment Type, Lead Time, Appointment Weekday, Patient Age, Reminder Message, and Patient Gender. The performance measures obtained for Model 4 are as follows: Accuracy–0.93, Precision–0.80, Specificity–0.94, Recall–0.91, F-measure–0.84, MCC–0.81, Log Loss–2.40, and AUC–0.98. Reciprocal Ranking was utilized to select 8 variables for prediction in Model 5. The variables used for prediction in Model 5, employing Reciprocal Ranking, include Missed Appointment History, Lead Time, Patient Trust, Patient Age, Appointment Weekday, Appointment Type, Reminder Message, and Handicap. The performance measures obtained for Model 5 are as follows: Accuracy–0.93, Precision–0.83, Specificity–0.96, Recall–0.85, F-measure–0.84, MCC–0.80, Log Loss–2.41, and AUC–0.98. Thus, all the models are seen to have performed well in predicting the appointment no-shows using Multilayer Perceptron. The results shows that that models with Univariate Feature Selection, Recursive Feature Elimination, Random Forest Classifier, and Reciprocal Ranking performed better with fewer features. The most critical features consistently across all models were Patient Trust, Appointment Type, Reminder Message, Lead Time, and Missed Appointment History. The variable Diabetics was not selected by any of the feature selection techniques used in the study. The variables used by different models are given in Table 3.

**Table 3** Variables used by different models

Variables	Model 1	Model 2	Model 3	Model 4	Model 5
Patient age	✓	✓	✗	✓	✓
Patient gender	✓	✗	✗	✓	✗
Appointment type	✓	✓	✓	✓	✓
Missed appointment history	✓	✓	✓	✓	✓
Patient trust	✓	✓	✓	✓	✓
Appointment week day	✓	✗	✗	✓	✓
Lead time	✓	✓	✓	✓	✓
Reminder message	✓	✓	✓	✓	✓
Scholarship	✓	✓	✓	✗	✗
Hypertension	✓	✓	✓	✗	✗
Diabetics	✓	✗	✗	✗	✗
Alcohol addiction	✓	✗	✓	✗	✗
Handicap	✓	✗	✗	✗	✗

## 5 Conclusion

The study aimed to compare the performance of different neural network models in predicting appointment no-shows using different feature selection methods. The results indicated that models with Univariate Feature Selection, Recursive Feature Elimination, Random Forest Classifier, and Reciprocal Ranking performed better with fewer features. These results have significant implications for researchers in identifying the most important features for predicting appointment no-shows. By embracing the most effective features, researchers can enhance predictive performance while utilizing fewer variables, ultimately leading to a reduction in the complexity of the prediction model. Therefore, this study contributes to the ongoing research on appointment no-shows and provides a useful reference for researchers to develop more accurate and efficient predictive models. The prospects of this research include the potential application of the findings in clinical settings to assist in mitigating the frequency of no-shows for outpatient appointments. This could ultimately lead to better resource utilization and improved patient outcomes. The current study has considered missed appointments and cancelations together; the study can be enhanced by considering missed appointments and cancelations separately. Tardiness can also be included to develop a better model. Another recommendation is to address the imbalance problem [28]. The dataset used in the study is imbalanced, with appointment shows comprising 80% of the dataset, while no-shows account for only 20%. However, the current study did not implement any strategies to rectify the imbalance problem. Future studies can consider employing methods like SMOTE to tackle the imbalance issue [29]. The outcome of the study helps the researchers to better understand about the predictors of hospital missed appointments.

## References

1. Uzosike U, Kangal E (2018) Dimensionality reduction in predictive analytics: filter-based feature selection approach. *Procedia Comput Sci* 126:244–252
2. Suruliandi A, Mariammal G, Raja SP (2021) Crop prediction based on soil and environmental characteristics using feature selection techniques. *Math Comput Model Dyn Syst* 27(1):117–140
3. Liu H, Motoda H (eds) (2012) Feature selection for knowledge discovery and data mining, vol 454. Springer Science & Business Media
4. Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. *J Mach Learn Res* 3(Mar):1157–1182
5. Erdem E, Bozkurt F (2021) A comparison of various supervised machine learning techniques for prostate cancer prediction. *Eur J Sci Technol* 21:610–620
6. Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning: data mining, inference, and prediction, 2nd edn. Springer
7. Chen J, Zhang H (2008) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans Pattern Anal Mach Intell* 30(8):1226–1238
8. El Mrabti S, Al Achhab M, Lazaar M (2018) Comparison of feature selection methods for sentiment analysis. In: Tabii Y, Lazaar M, Al Achhab M, Enneya N (eds) Big data, cloud and

- applications, BDCA 2018. Communications in computer and information science, vol 872. Springer, pp 221–234
- 9. Mohan S, Thirumalai C, Srivastava G (2019) Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access* 7:81542–81554
  - 10. Liu S, Jiang Q, Ma Y, Xiao Y, Li Y, Cui C (2017) Object-oriented wetland classification based on hybrid feature selection method combining with relief f, multi-objective genetic algorithm, and random forest. *Nongye Jixie Xuebao/Trans Chin Soc Agric Mach* 48(1):119–127
  - 11. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay É (2011) Scikit-learn: Machine learning in Python. *J Mach Learn Res* 12:2825–2830
  - 12. Liu H, Yu L (2005) Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans Knowl Data Eng* 17(4):491–502
  - 13. Li Y, Huang D, Liang Y (2019) An ensemble feature selection method for cancer classification based on gene expression data. *BMC Bioinform* 20(1):1–18
  - 14. Effrosynidis D, Arampatzis A (2021) An evaluation of feature selection methods for environmental data. *Eco Inform* 61:101224
  - 15. Kavitha CR, Mahalekshmi (2017) Feature selection methods for classification: A comparison. *Int J Res Eng Technol (IJRET)* 6(06):222–226
  - 16. Spencer R, Thabtah F, Abdelhamid N, Thompson M (2020) Exploring feature selection and classification methods for predicting heart disease. *Digit Health* 6
  - 17. Yang Y, Li Y, Zhang X (2021) Comparing different feature selection algorithms for cardiovascular disease prediction. *J Med Syst* 45(3):1–9
  - 18. Alshammari R, Daghstani T, Alshammari A (2020) The prediction of outpatient no-show visits by using deep neural network from large data. *Int J Adv Comput Sci Appl* 11(10)
  - 19. Liu D, Xu Y, Lu C, Lin J, Li M, Ma S (2022) Machine learning approaches to predicting no-shows in pediatric medical appointment. *NPJ Digit Med* 5(1):50
  - 20. Dashtban M, Li W (2019) Deep learning for predicting nonattendance in hospital outpatient appointments. In: Proceedings of the 52nd Hawaii international conference on system sciences (HICSS), pp 3731–3740
  - 21. Mohammadi I, Wu H, Turkcan A, Toscos T, Doebling BN (2018) Data analytics and modeling for appointment no-show in community health centers. *J Prim Care Community Health* 9:2150132718768011
  - 22. Nelson A, Herron D, Rees G, Nachev P (2019) Predicting scheduled hospital attendance with artificial intelligence. *NPJ Digit Med* 2(1):1–7
  - 23. Fan G, Deng Z, Ye Q, Wang B (2021) Machine learning-based prediction models for patients no-show in online outpatient appointments. *Data Science and Management* 2:45–52
  - 24. Alshaya S, McCaren A, Al-Rasheed A (2019) Predicting no-show medical appointments using machine learning. In: Communications in computer and information science. Springer International Publishing, pp 211–223
  - 25. Turner CR, Fuggetta A, Lavazza L, Wolf AL (1999) A conceptual basis for feature engineering. *J Syst Softw* 49(1):3–15
  - 26. Bradley AP (1997) The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recogn* 30(7):1145–1159
  - 27. Chicco D, Gurman G (2020) The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics* 21(6)
  - 28. Fan W-W, Lee C-H (2021) Classification of imbalanced data using deep learning with adding noise 1–18
  - 29. Wang S, Dai Y, Shen J et al (2021) Research on expansion and classification of imbalanced data based on SMOTE algorithm. *Sci Rep* 11:24039

# A Simple Recommendation Model Using the Item's Global Popularity and Frequency-Based User Preference



Somaraju Suvvari and Md Iftekhar Ahmad

**Abstract** Next-basket recommendation (NBR) is one of the emerging methods in the recommendations system and it has many real-world applications, like where you want to predict the set of items in the next transaction based on its previous continuous transactions. In the literature, many algorithms are proposed for the next-basket recommendation, but very little attention is given to the frequency information in making the recommendation model. In this paper, we presented a simple recommendation model, and in this model, the user representation is done based on the frequent purchasing information, and finally, when the recommendation is made, it considers the user representation and items global support. We made experiments on publicly available three real-time datasets and compared them with two of the existing recommendation algorithms using recall as a measure, and our algorithm outperforms the existing two models. The proposed algorithm is simple in terms of its complexity and the number of operations involved.

**Keywords** Next-basket recommendation · User general preference · Collaborative filtering

## 1 Introduction

It is inevitable that in today's real world, no one can live without technology, and technology is growing very rapidly as time progresses. One area where technology is growing very rapidly is e-commerce, and due to its development in e-commerce, a very huge number of products are available for the same purpose. Due to the huge availability of products, it is not at all possible for the user which product(s) is to be selected, as it is not possible to go through the description of each product in reality.

---

S. Suvvari (✉) · M. I. Ahmad  
National Institute of Technology Patna, Patna, India  
e-mail: [somaraju@nitp.ac.in](mailto:somaraju@nitp.ac.in)

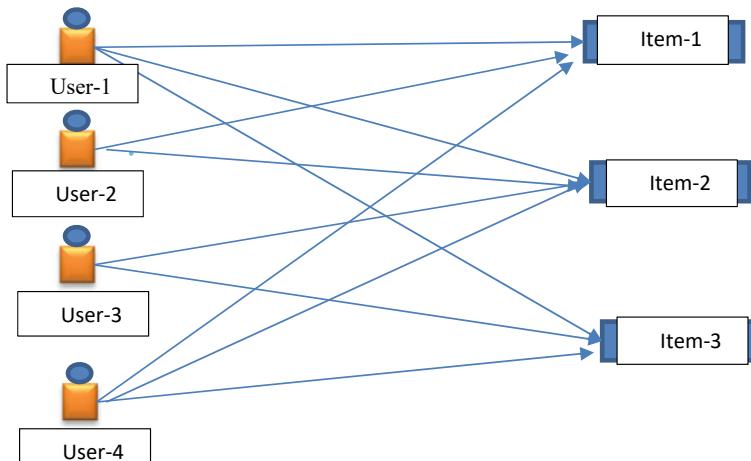
M. I. Ahmad  
e-mail: [mda.phd20.cs@nitp.ac.in](mailto:mda.phd20.cs@nitp.ac.in)

In most cases, the user is unaware of the different products for the same purpose. This is the scenario where the recommendation system helps the user to select the proper product. The recommendation model understands the taste of the user based on his past history and it suggests suitable objects to the user.

In literature, based on the work, the recommendation system algorithms are categorized into two types, collaborative recommendation algorithms' [1–3] system and content-based recommendation algorithms. In the collaborative recommendation system algorithms, the user's history is observed, and to find similar users, in the case of user-based collaborative filtering, suggest the items to the target user that were in similar users' transactions but not in the target user. In the case of an item-based collaborative recommendation system, the algorithm tries to find the items which are more similar and suggests the target item to the users rated/liked the similar items to the target item. The illustrative example of a collaborative filtering algorithm is shown in Fig. 1.

In Fig. 1, for example, if user-2 is the target user, then consider the user-based collaborative filtering algorithm, then find similar users to user-2. Here user-1 and user-4 (since they share two out of three items) are similar users to user-2, so item-3 will be recommended to user-2. In the content-based recommendation system, the algorithm maintains the profiles of items and users, and it recommends the items which match with the user profile; mostly, it recommends new or unpopular items.

One way of recommending the items using the following principle: "Consider that the target user has purchased a kind of product, can we recommend similar products to the products purchased in the past history". The collaborative recommendation algorithms generally follow this principle. Most of the attention is given to this principle in the recommendation literature. The other principle is that: "Can we recommend the products/items which the target user has purchased in the past". We call this as repetitive recommendation principle. Very little attention has been



**Fig. 1** Collaborative filtering algorithm example

given to this principle in the recommendation literature. But, this repetitive recommendation principle cannot be ruled out in most of the applications. For example, most of the consumable items like bread, butter, toothpaste, etc., will be repeated in the transactions. We also observed the same phenomena in other applications like music listening (where a customer may listen to the same song repetitively), the stock market industry (where a customer buys and sells the same company stocks in different transactions), etc. Another observation we made is that in these types of applications, some products present in the user transactions belong to the category of popular products. So, we concluded that there must be some approach that combines the repetitive recommendation principle and item global support. As per our observation, we were the first to propose an algorithm using the repetitive recommendation principle and item global support.

In this report, we present a model which recommends the items which are more repeatedly purchased and the items which have more popularity, and the novelty of our algorithm is that both the properties are measured using the support measure only. We have done the experiments on three different publicly available datasets. The remaining report is organized as per the following: Sect. 2 discusses the related works in the recommendation system; Sect. 3 discusses the proposed approach; Sect. 4 discusses the experimental results; and Sect. 5 discusses our conclusion.

## 2 Related Work

In the literature, there exist many works related to recommendation algorithms [1, 4–10]. Next-basket recommendation grabbed the attention of researchers in recent years. In the earlier part of the literature, most of the works were based on the Markov chain property. The Markov chain has the property to extract the sequential nature from the dataset. Steffen Rendle et al. [11] presented a model based on the Markov chain property, and the authors tried to combine both the Markov chain method and Matrix Factorization methods to get the advantages of both general recommendation and sequential recommendation. But, the algorithm combines both properties linearly, so lacks in exhibiting the nonlinear nature.

Pengfei Wang, et al. in [12], proposed an algorithm to overcome the limitation in the approach in [11]. In [12], the authors combine the different factors with the assumption that the factors may be dependent which is lacking in [11]. In their proposed approach, the authors proposed multilayer architecture to get the hybrid representation of the user using the user representation and item representation from the last transaction and try to predict the user purchasing items in the next transaction.

Feng Yu, et al. [13] proposed a method based on a recurrent neural network, where they succeeded in combining the different factors nonlinearly. They encoded the baskets and represented each user with a vector, and finally, they predicted the score of each item in his/her next basket.

Duc-Trong Le, et al., [14] proposed an approach based on deep learning approach, where the authors introduced a concept called correlation matrix which records the

relation strength between two items, and the encoded basket sequence and the LSTM network take these two as an input and it tries to predict the score for the items in the next basket.

Bo Peng, et al., [15] proposed an approach for a sequential recommendation, they use the user's general preference, and in the recent transactions of the user, they extracted the high-/low-order association patterns and the synergies between the items. The entire proposed architecture works in three phases, in the first phase, the users and items will be embedded, the second layer named as pooling layer tries to extract the relation between the items in history and target items, and finally, the third layer called prediction layer tries to predict the items in the next transaction.

There exists very little work based on the frequent information of the items [16–19]. The authors in [16] use frequency information in recommending the items. The authors build a user vector of size equal to the number of items in the dataset, and this user vector holds the information related to the frequency information. Next, the authors find the k-most similar users using the KNN classifier and averaging the respective items' information to make a vector which is the user's neighbor representation. In the prediction, the authors use both of these information. The authors proved that the frequency information cannot be captured properly using the deep learning approaches.

### 3 Proposed Methodology

#### 3.1 Formalizations

Given the set of transactions made by all the users  $h_1, h_2, h_3, \dots, h_n$ , where  $h_1$  is the set of all baskets of user1,  $h_2$  is the set of all baskets of user2, ...,  $h_n$  is the set of all baskets of user n. Build a recommendation system to recommend the items in the next basket based on the user's preference in terms of repeatedly purchasing items and the item's global popularity.

In the literature, researchers proposed some methods for finding repeatedly purchasing items. For example, in [16], the authors represented each basket with a size equal to the number of items in the dataset, and by applying the decaying weight on each item respective to the position of the basket and by averaging them, the authors get a vector which represents the user and inherently the value of each item in the vector represents the importance of its frequent occurrence in the transactions. In this report, we are coming up with a different approach, where our intention is that we should give more importance to the items where the item's support is increasing as time progresses, and we should give less importance to the items where the items' support is decreasing as the time goes on.

### 3.2 Model

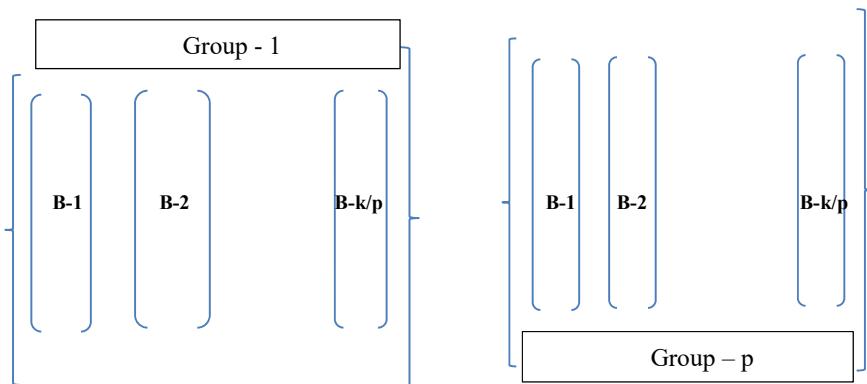
Our model first finds the two components; one user's preference in terms of his interest in the items which frequently purchased another one is the item's global popularity. We combine these two components to recommend the items in the next basket.

#### Finding the User Preference

Here, we consider the transaction of each user as a binary vector of size equal to the number of transactions, where if the item is present, its value in the vector is one in the respective position, and if the item does not present, then its value will be zero. For example, the item set size is 5, and if the user has made a transaction by purchasing item-1 and item-3, then the vector should be like this: (1,0,1,0,0). Through this report, we use  $m$  for the number of items,  $n$  for the number of users, and  $k$  for the number of baskets for each user. The following is the approach we followed to get the user preference:

**Step 1:** Group the  $k$  baskets of user  $j$  into  $p$ -groups (where  $p$  is the user-defined threshold, and the last basket may hold more baskets) which is shown in Fig. 2. In each group, there exist  $k/p$  baskets. In Fig. 2, it has been shown as  $B-1$  is the basket one,  $B-2$  is the basket 2 ...  $B-k/p$  is the last basket.

**Step-2:** In each group, find the support of each item with respect to the group size. Let us consider the support of all the items in group 1 is represented by vector  $v_1$ , group 2 is  $v_2$ , group 3 is  $v_3$ , ..., in group  $p$  is  $v_p$ . Initially, the user  $j$ 's vector  $U$  is empty. Find the difference between  $v_2$  and  $v_1$ , and add it to  $U$ . If the value of the respective item in  $U$  is positive, means the user frequently purchases the respective item (the support in group 2 is more than group 1), if the value is negative means the popularity of the item in group 2 is less than group 1 and it means the user is not showing interest in the respective item, make the value to zero for the items whose



**Fig. 2** User  $j$ 's  $p$ -groups

value is negative. Do this process for all the consecutive groups. Finally, the resultant vector U represents the user.

**Finding the item's global popularity:** We use the following way to calculate the item popularity: If an item is present in at least one basket of the user, then that item's popularity will be incremented by one and it is done for all the items. Finally, we divide this by the number of customers. We make a vector S to represent item popularity.

### Prediction

The final prediction vector P of items in the next basket is determined by using user representation U and the support vector S, and it uses the following equation:

$$\mathbf{P} = \alpha \cdot \mathbf{U} + (1 - \alpha) \cdot \mathbf{S} \quad (1)$$

Here,  $\alpha$  is user-defined hyperparameter and it represents the importance given to the parameter U.

## 4 Experimental Results

To evaluate our performance, we conducted experiments on the following three datasets which were preprocessed by the authors in [16], and their characteristics after preprocessing are shown in Table 1.

**Evaluation Metric:** To find the efficiency of our model, the evaluation metric we used is recall@k and is defined in the following way:

$$R@k = \frac{\text{Number of } r \text{ relevant items captured by } k \text{ recommended items}}{\text{Total number of relevant items}}. \quad (2)$$

We compared our results with two state-of-the-art methods, userKNN [20] and RepeatNet [19]. We can conclude from the results shown in Table 2 that the method discussed in this report which combines the user's frequent information with the items global support performs better than the existing non-deep learning method userKNN [20]. Despite being a non-deep learning method, its performance is comparable to

**Table 1** Dataset characteristics

Dataset	Total users	Total items	Average transaction size	Average number of transactions per user
Instacart <sup>a</sup>	19,935	7999	8.87	7.97
Dunnhumby <sup>a</sup>	36,241	4995	7.60	4.00
TaFeng <sup>a</sup>	13,949	11,997	6.22	5.69

<sup>a</sup> GitHub—HaojiHu/TIFUKNN: kNN-based next-basket recommendation; we accessed it on 08-05-2023

**Table 2** Comparison of results with state-of-the-art methods

Dataset/metric	userKNN [20]		RepeatNet [19]		Proposed approach	
	R@10	R@20	R@10	R@20	R@10	R@20
Instacart	0.0720	0.1260	0.2107	0.2637	<b>0.24181</b>	<b>0.3095</b>
Dunnhumby	0.1135	0.1648	0.1324	0.1989	<b>0.16</b>	<b>0.213</b>
Tafeng	0.1089	0.1278	0.0645	0.0919	<b>0.1180</b>	<b>0.1577</b>

the deep learning method RepeatNet [19]. In our experiments, we used *alphavalue* 0.7, and in the Instacart dataset, the number of groups is 3; in the Tafeng dataset, the number of groups is 4, and in Dunnhumby dataset, the number of groups is 3. It is evident from Table 2 that the proposed algorithm is giving better results compared to the other algorithms. The complexity of our model is simpler than the existing models and its performance is also better than most of the existing models. In any application where the dataset is more complicated, in terms of the number of users, items, and transactions, then our proposed model can be preferred compared to the complex deep learning models.

## 5 Conclusion

In this report, we proposed a simple algorithm for next-basket recommendation using the user preference in terms of repeated purchasing behavior of the user and the item's global support. The proposed algorithm is simple in terms of its complexity and the number of operations involved. The algorithm gives better results compared to two of the existing algorithms on three different datasets.

## References

- He X, Liao L, Zhang H, Nie L, Hu X, Chua T-S (2016) Neural collaborative filtering. In: Proceedings of the 26th international conference on World Wide Web. International World Wide Web conferences steering committee, pp 173–182
- Liang D, Krishnan RG, Hoffman MD, Jebara T (2018) Variational autoencoders for collaborative filtering. In: Proceedings of the 2018 World Wide Web conference on World Wide Web. International World Wide Web conferences steering committee, pp 689–698
- Ma C, Kang P, Liu X (2019) Hierarchical gating networks for sequential recommendation. In: Proceedings of the 25th ACM SIGKDD International conference on knowledge discovery & data mining, pp 825–833
- Hu W, Pennington J (2020) Provable benefit of orthogonal initialization in optimizing deep linear networks. In: International conference on learning representations, London, 2020
- Lian J, Zhou X, Zhang F, Chen Z, Xie X, Sun G (2018) xDeepFM: combining explicit and implicit feature interactions for recommender systems. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining (KDD'18). ACM, pp 1754–1763

6. Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans Knowl Data Eng* 17(6):734–749
7. He X, Deng K, Wang X, Li Y, Zhang Y, Wang M (2020) LightGCN: simplifying and powering graph convolution network for recommendation. In: The 43st international ACM SIGIR conference on research & development in information retrieval, pp 639–648
8. Horasan F (2022) Latent semantic indexing-based hybrid collaborative filtering for recommender systems. *Arab J Sci Eng* 47:10639–10653
9. Roy D, Dutta M (2022) A systematic review and research perspective on recommender systems. *J Big Data* 9:59
10. van Maasakkers L, Fok D, Donkers B (2023) Next-basket prediction in a high-dimensional setting using gated recurrent units. *Expert Syst Appl* 212
11. Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized Markov chains for next-basket recommendation. In: Proceedings of the 19th international conference on World Wide Web (WWW'10), pp 811–820
12. Wang P, Guo J, Lan Y, Xu J, Wan S, Cheng X (2015) Learning hierarchical representation model for next basket recommendation. In: SIGIR, pp 403–412
13. Feng Y et al (2016) A dynamic recurrent model for next basket recommendation. In: Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval. ACM
14. Le DT, Lauw HW, Fang Y (2019) Correlation-sensitive next-basket recommendation. In: Proceedings of the 28th international joint conference on artificial intelligence, Macau, pp. 2808–2814. Research Collection School of Information Systems, China
15. Peng B, Ren Z, Parthasarathy S (2022) M2: mixed models with preferences, popularities and transitions for next-basket recommendation. *IEEE Trans Knowl Data Eng* (TKDE)
16. Hu H, He X, Gao J, Zhang Z-L (2020) Modeling personalized item frequency information for next-basket recommendation. In: ACM DL or in arXiv. In the 43th International ACM SIGIR conference on research and development in information retrieval, 2020
17. Hu H, He X (2019) Sets2Sets: learning from sequential sets with neural networks. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 1491–1499
18. Wang C, Zhang M, Ma W, Liu Y, Ma S (2019) Modeling item-specific temporal dynamics of repeat consumption for recommender systems. In: The World Wide Web Conference, pp 1977–1987
19. Ren P, Chen Z, Li J, Ren Z, Ma J, de Rijke M (2019) RepeatNet: a repeat aware neural recommendation machine for session-based recommendation. In: Proceedings of the AAAI conference on artificial intelligence, vol 33, pp 4806–4813
20. Konstan JA, Miller BN, Maltz D, Herlocker JL, Gordon LR, Riedl J (1997) GroupLens: applying collaborative filtering to Usenet news. *Commun ACM* 40(3):77–87

# An Early Detection of Autism Spectrum Disorder Using PDNN and ABIDE I&II Dataset



Manjunath Ramanna Lamani<sup>✉</sup> and P. Julian Benadit<sup>✉</sup>

**Abstract** The current study's objective was to use deep learning methods to separate valetudinarians amidst autism spectrum disorders (ASDs) from controls employing just the patients' brain activation patterns from a dataset of large brain images. We examined brain imaging data from ASD patients from the global, multi-site ABIDE dataset (Autism Brain Imaging Data Exchange). Social impairments and repetitive behaviors are hallmarks of the brain condition known as autism spectrum disorder (ASD). ASD affects one in every 68 kids in the USA, as of the most recent data from the Disease Control Centers. To understand the neurological patterns that arose from the categorization, we looked into functional connectivity patterns that can be used to diagnose ASD participants precisely. The outcomes raised the state of the art by correctly identifying 72.10% of ASD patients in the sample vs. control patients. The classification patterns revealed an anti-correlation between the function of the brain's anterior and posterior regions; this anti-correlation supports the empirical data currently showing achingly ASD impedes communication between the livid brain's anterior and posterior areas. We found and pinpointed brain regions damn frolic, distinguishing ASD among typically developing reign according to our deep learning model.

**Keywords** ASD · fMRI · ABIDE I · ABIDE II · rsfMRI · DL · ML

## 1 Introduction

Research in psychiatric neuroimaging primarily focuses on locating reliable biomarkers that could help prognosis and manage brain-based diseases. A promising strategy for examining the reproducibility pattern function of brain across more giant, more homogeneous datasets is data-intensive machine learning [1]. Using the dataset from resting state functional magnetic resonance imaging (rs-fMRI), the primary

---

M. R. Lamani (✉) · P. Julian Benadit

CHRIST (Deemed to be University), Kanmanike, Kumbalgudu, Mysor Road, Bangalore, Karnataka 560074, India

e-mail: [manjunath.lamani@res.christuniversity.in](mailto:manjunath.lamani@res.christuniversity.in)

objective of the current investigation was to categorize participants with autism spectrum disorders (ASD) and controls particular brain specimen Spartan kinship. Unsupervised and supervised machine learning (ML) techniques were merged in a deep learning approach that we utilized. In order to test the technique on a significant sample group of brain imaging data, Autism Imaging Data Exchange-I&II (ABIDE I&II). Subsequent goal investigated brain motif connected to ASD which was crucial for classification.

Researchers findings examined in relation to other studies on brain activity of ASD sufferers as well as areas of brain connections that set ASD apart from controls. Different phenotypes of ASD are linked to social, communication, and sensorimotor deficiencies, range in severity, linguistic, and sensorimotor impairments. ASD diagnostic tools evaluate recognizable social traits and linguistic abilities. However, the complexity of the range of behavioral disorders in autism may be precisely mapped against their brain patterns, thanks to neuroscientific researchers, which has reduced the flous [2]. Noninvasive brain imaging studies have enhanced our knowledge of the neural underpinnings of brain disorders and the behavior that goes along with them, namely ASD and the endemic and communication challenges that accompany it [3–6]. To better understand the causes of mental diseases, crimped impetus for ASD has been identified, these patterns have been linked to neurological and psychosocial factors [7]. Replicating results across more extensive, more diverse datasets that represent the diversity of clinical populations present a difficulty for brain imaging research of brain illnesses. To uncover repeatable patterns of brain function, ML tactics used to brain imaging data. Algorithms derive reliable, rugged neural moire brain fiction specifics people with loco disorders [8].

## **2 The Next Step in Comprehending the Brain and Psychiatric Diseases is Machine Learning, Deep Learning, and Predicting Disease States**

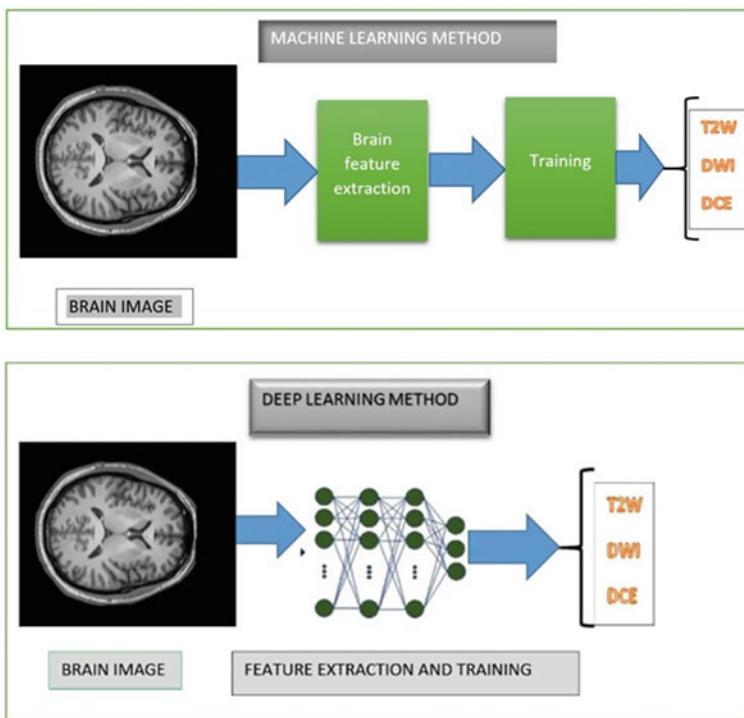
Classification emotional plight related to delineation semiotic genre, noun meanings, emotions, and learning has been made possible by combining ML techniques with brain imaging data [9–14]. Studies on patients with mental disease states have discovered some who have brain activation linked to sadness, autism, and schizophrenia [15, 16]. The accuracy of classifying singles as autism or normal based on fMRI brain vitalize 97% by researchers adopted ML tactics ASD brain ascetic data. Brain swatch activity linked to subconscious element. Designed nonexistent autistic participants but prevalent in control patients [4]. In a different study [17], the authors classified participants with ASD with the accuracy is 76.67% using a total patients sample of 179 persons with ASD and participants with similar IQs who were usually developing.

In most studies combining brain imaging as well as ML, supervised learning techniques like support vector machines (SVM) or Gaussian naive Bayes (GNB) classifiers have been used. Subjective nature feature selection processes for supervised ML tactics could make it difficult to compare the outcomes of different experiments. In supervised methods, group data used set of training data is given class labels, and a different points of dataset is categorized following the patterns seen training data. These labels and attributes are chosen based on a priori hypotheses or exploratory techniques; as a result, they depend on a degree of subjectivity. Using sets of 100, 200, 400, and more voxels as an instance, size in voxels worn for classifying brain fiction data pragmatic chosen to determine appropriate classification set size [11, 12].

Current swotting, we deal unrealizable problems employing sizable data immovable also unsupervised ML technique. We categorize a psychiatric condition. Feature extraction for eliminating the subjects might open up a nova, more data-driven insight into how the brain functions. Figure 1 describes the sequence identification in multi-parametric function magnetic resonance imaging (mp-FMRI) using machine learning versus deep learning. Through feature extraction unique to the various sequences of T2-weighted imaging (T2W), diffusion-weighted imaging (DWI), and dynamic contrast-enhanced imaging, the computer can learn from inputs of mpMRI pictures (DCE). Following training on further photos, the computer can recognize the right order as an output. In contrast to machine learning, deep learning allows for simultaneous feature extraction and training to get the desired results.

### 3 ABIDE Dataset Classification

ABIDE I and II data were used to categorize autism versus healthy participants using assessments of brain connectivity. We have downloaded ABIDE I and II datasets from the Neuroimaging Tools and Resources Collaboratory (NITRC) and Neuroimaging data repository (link for datasets download: [https://fcon\\_1000.projects.nitrc.org/indi/abide/](https://fcon_1000.projects.nitrc.org/indi/abide/)). The authors modified a technique described in [18] and replicated it using data from other sites. For the 964 participants analyzed, the BOLDs signal from nonoverlapping gray scale ROIs (new SPM12 and Free surfer tools) created beyond seed defocus spaced alongside computed 5 mm. Voxels that were Euclidean-near to the seed voxel of a specific ROI were included in this ROI. By determining the pair-wise correlation between each ROI and the data from the 7266 produced ROIs. To correlate the connection vector matrix with subjects-related characteristics like phynotypes data, standard linear approach had fitted exclusive group (autism/normal) using a leave\_one\_out method. Based on the factors, an estimation of each connection's value for the excluded subject was made. The difference between the means for the connection on two different sites was then used to make the necessary corrections. This technique reduced between-site variables, like various scanners and variations in parameters scanning and processes, that could skew results.



**Fig. 1** ML preprocessing steps and DL preprocessing steps with mp-FMRI

#### 4 Deep Learning Algorithms and Imaging of the Brain

Deep neural networks were used by [19] to examine brain actions from measured brain activities. Researchers employed fMRI task-based data taken away 499 people to train artificial neural networks (ANN) with duplet veiled layers, a softmax output layer, and duplet veiled classify data into five task-related categories: vehemence, language, relationship, endemic, also functional evocation. Unlike supervised learning techniques like Linear Regression (LR) as well as SVM, which had a mean accuracy of 47.97%, Deep Models produced higher results (mean accuracy of 50.74%) [14]. Data from four distinct sites were used to distinguish individuals schizophrenia vs mated hale juice using structural T1-weighted images and deep learning. The PREDICT-HD project's data were used by the authors to classify individuals with Huntington's disease in comparison to healthy controls. Classified 191 control participants with 198 schizophrenia patients emerge 4 separate swotting conducted by different university. Deep Belief Network (DBN), train by different layers(50, 50, and 100 hidden units are in first, second and top layer). In comparison to classification ACC for SVM-68% using raw data, classification ACC 98% employing

features taken beyond 3 DBMs. According to scientists, DL holds enormous potential for applications in diagnostic brain imaging [20].

## 5 Resources and Techniques

### 5.1 Participants

ABIDE I&II provided rs-fMRI data used in this investigation. The consortium provided the rs-fMRI ASD and mated reign data ABIDE for aim of sharing data in scientific community [21] and visit:). ABIDE I Data from 539 people with ASD and 553 matched controls were included (typical controls, TC) and ABIDE II dataset from 521 people with ASD and 588 matched controls were included (TC). ABIDE I and II datasets, gathered 20 and 19 various imaging locations, comprise rs-fMRI pictures, brain images of T1 structural also phenotypic data for exclusive patient. A summary of these datasets is shown in Tables 1, 2 and 3.

**Table 1** ABIDE-I dataset summary

Universities	Total number of participants	ASD	Typical controls	Age (years)
1_CIT	38	19	19	17–56.2
2_CMU	27	14	13	19–40
3_KKI	55	22	13	8–12.8
4_LMUM	57	24	33	Jul-58
5_NYULMC	184	79	105	6.5–39.1
6_OILHH	36	20	16	Oct-24
7_OH&SU	28	13	15	8–15.2
8_SDSU	36	14	22	8.7–17.2
9_SBL&BCN&NIC&UMCG&NIN	30	15	15	20–64
10_SU	40	20	20	7.5–12.9
11_TCHS	49	24	25	12–25.9
12_UCLAS1	82	49	33	8.4–17.9
13_UCLAS2	27	13	14	9.8–16.6
14_ULS1	29	14	15	18–32
15_ULS2	35	15	20	12.1–16.9
16_UMS1	110	55	55	8.2–19.2
17_UMS2	35	13	22	12.8–28.8
18_UPSM	57	30	27	9.3–35.2
19_UUSM	101	58	43	8.8–50.2
20_YCSC	56	28	28	7–17.8

**Table 2** ABIDE-II dataset summary

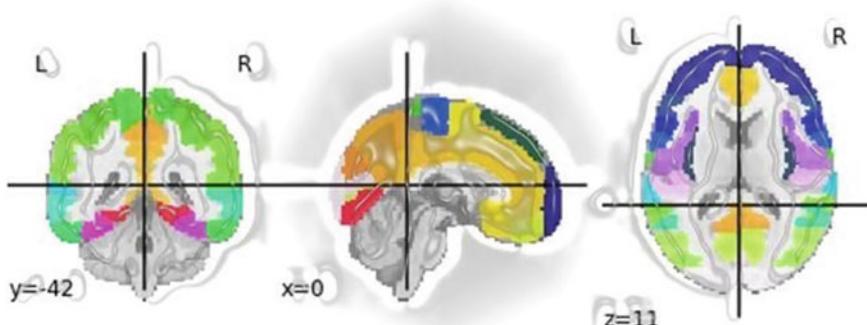
Universities	Total number of participants	ASD	Typical controls	Age (years)
<i>Details of ABIDE II dataset</i>				
1_BNI	58	29	29	18–64
2_EUMCR	54	27	27	06-Nov
3_ETHZ	37	13	24	14–31
4_GU	106	51	55	8.1–13.9
5_IU	40	20	20	17–54
6_IP&RDH	56	22	34	Jun-47
7_KUL	28	28	—	18–35
8_KKI	211	56	155	Aug-13
9_NYULMC_S1	78	48	30	5.2–34.8
10_NYULMC_S2	27	27	—	5.1–8.8
11_ONRC&ILHH	59	24	35	18–31
12_OH&SU	93	37	56	Jul-15
13_TCHS	42	21	21	Oct-20
14_SDSU	58	33	25	7.4–18
15_SU	42	21	16	Aug-13
16_UCD	32	18	14	Dec-18
17_UCLA	32	16	16	Aug-15
18_UM	28	13	15	Jul-13
19_UUSM	33	17	16	Jul-13

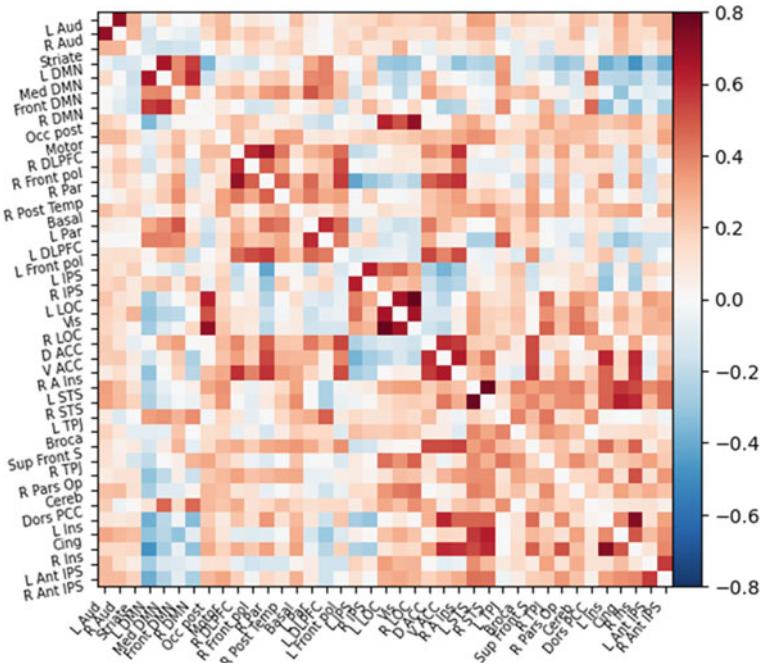
## 5.2 Resting State and Feature Selection

The functional link between different brain regions can be measured neurally using resting state fMRI. For studying clinical populations, Rs-fMRI data is beneficial. It enables research into disrupted brain networks without the complexity and diversity that come with task-related brain activation [17]. It can be applied to examine mental states, among other things, memories and event recall, and clinical populations [21, 22]. It has been demonstrated that Rs-fMRI is highly repeatable and that it produces datasets that are simple to compare between investigations [22, 23]. Low-frequency oscillation in blood oxygenation causes correlation oscillation rsfMRI. It is an example of the brain's functional connection [24]. A correlation is generated for moderate time series of ROIs to study brain connection. The connectivity matrix is constructed using the correlation. Figure 2 describes the different regions of interest in the brain atlas from the FSL and specifies the different axis ( $x$ ,  $y$ , and  $z$ ) it determines the different annotation with neuroimaging plotting with Nilearn displays function (displays the ortho slicer) Fig. 3. Describes the correlation of different brain different connectivity matrix with time series. Figure 4 represents the other brain region of interests coordinates with different intensities (correlation matrix).

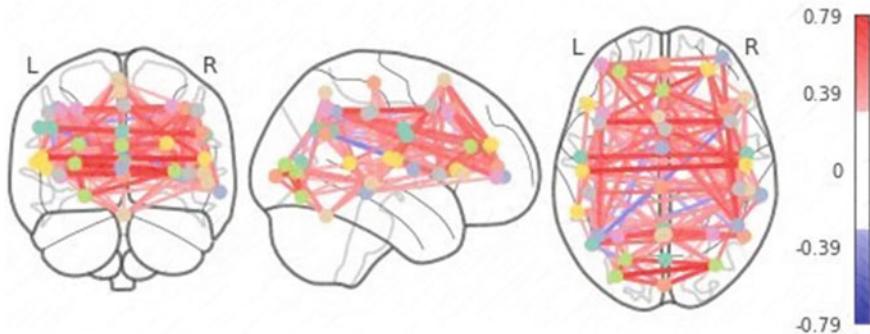
**Table 3** Leave\_site\_out fivefold cross\_validation outcome employing PDNN-ABIDE-I

Universities	Total number of participants	ACC	SENS	SPEC
<i>Details of ABIDE I dataset</i>				
1_CIT	38	0.65	0.61	0.71
2_CMU	27	0.64	0.6	0.7
3_KKI	55	0.63	0.62	0.62
4_LMUM	57	0.62	0.68	0.68
5_NYULMC	184	0.66	0.67	0.67
6_OILHH	36	0.68	0.69	0.69
7_OH&SU	28	0.69	0.62	0.64
8_SDSDU	36	0.67	0.63	0.66
9_SBL&BCN&NIC&UMCG&NIN	30	0.63	0.61	0.62
10_SU	40	0.64	0.62	0.68
11_TCHS	49	0.64	0.62	0.7
12_UCLAS1	82	0.68	0.63	0.71
13_UCLAS2	27	0.64	0.68	0.69
14_ULS1	29	0.66	0.64	0.69
15_ULS2	35	0.62	0.65	0.67
16_UMS1	110	0.61	0.52	0.69
17_UMS2	35	0.68	0.59	0.67
18_UPSM	57	0.67	0.67	0.65
19_UUSM	101	0.64	0.68	0.68
20_YCSC	56	0.63	0.63	0.67
Mean	55.5	0.649	0.63	0.6745

**Fig. 2** Brain different regions of interest with three axes ( $x$ ,  $y$ , and  $z$ )



**Fig. 3** Correlation matrix



**Fig. 4** Brain region of interests coordinates

### 5.3 Data Preprocessing

It was possible to download previously processed rs-fMRI data from Preprocessed Connectomes Project's website (<http://preprocessed-connectomes-project.org/>). The C-PAC preparation pipeline was used to choose the data. Slice time, motion, and voxel intensity corrections adopted to rsfMRI data. Low-frequency drifts as well as

global signal as regressors, nuisance signal removal was carried out using 24 motion parameters, CompCor with five components, and low-frequency drifts. Functional data were band-pass filtered (0.01–0.1 Hz) and spatially registered to a template space using a nonlinear method (MNI152). Each subject was given the mean time series for ROIs. To minimize the size of the features vector, CC200 brain atlas functional parcellation [25] employed. This atlas was created by parcellating the entire brain, using data, into 200 geographically near regions with uniform functional activity.

## **5.4 Connectivity of Different Functions (Selections of Features)**

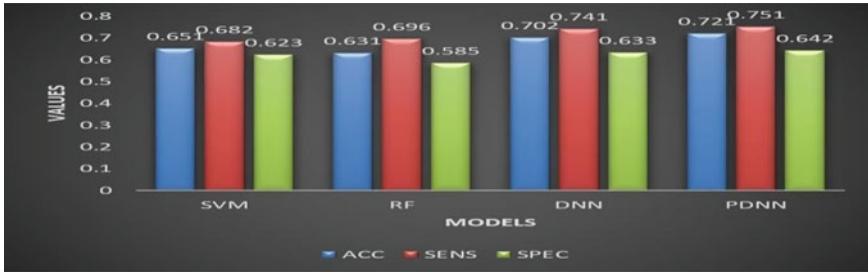
Subjects were categorized as having ASD or TC using ascetic kinship. Ascetic kinship provides an index of the level of coactivation of brain regions utilizing the series of time of rsfMRI neuroimaging data. A Pearson's correlation coefficient can be found in each cell of the connection matrix. The coefficient, which varies from 1 to 1, is a measure of the correlation between two brain regions. Values near 1 suggest that the time series are firmly linked, while values close to 1 indicate that the time series are anti-correlated. To use the values in the correlation matrix as features, the upper triangle values were subtracted. The values in the lower triangle are repeated in these values. Since the primary matrix diagonal depicts an area that correlates to itself, we also deleted it. Later, to obtain a vector of features for subject classification. The following Eq. 1 determines the number of resultant features:

$$s = \frac{(V - 1)V}{2} \quad (1)$$

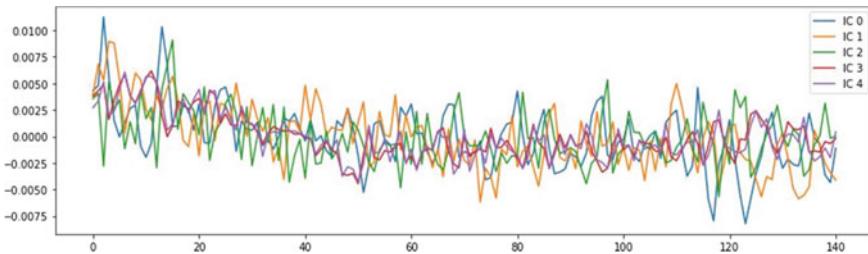
In Eq. 1, V represents connected number voxels/regions. The approach produced 19,900 features using the CC200 ROI atlas.

## **5.5 Classification Methods**

Autoencoders are trained unsupervised, one layer at a time. Encoder weights to multi-layer perceptron (MLP), assumes a 19,900 feature input space and a 2-number output space, as will be further discussed below. Between the input and output levels of the network are two hidden layers of 900 and 500 units respectively. Figures 1 and 2 contain the unsupervised trained encoders in green-blue weights—MLP in supervised-trained using prior information from autoencoder training. The supervised training of the MLP is referred known as fine-tuning since it uses weights that have been changed depending on the autoencoder encoders. Supervised task, to reduce prediction error and to produce the predicted classes, the MLP weights must be fine-tuned. Two output units, one for each output unit in the output layer, indicate



**Fig. 5** Comparison of proposed and existing models



**Fig. 6** Correlation of IC time series and Pearson's correlation between the components and the signal using by functional image

whether a given input is more likely to come from ASD/TC subject. One-hot outputs are those that are produced using a softmax function, with just one intended to have an activation value of 1 (and the rest, 0) during fine-tuning. The output distribution is normalized using softmax functions, producing outputs that indicate complementary probabilities of class membership. Figures 5 and 6. Comparison of Proposed Deep Neural Networks (PDNN), Deep Neural Networks (DNN), Random Forest (RF), and SVM classifiers trained employing cross-validation by 10 times on ABIDE I&II with different matrices (Accuracy (ACC), Sensitivity (SENS), Specificity (SPEC)).

## 5.6 Classifier Evaluation

The performance of the model was compared to the results of classifiers trained using SVM [26] and RF in order to assess the outcomes of deep learning [27]. All models are evaluated based on a tenfold cross-validation schema that combines data from all sites while maintaining the proportions among them. The encoder layers of an unsupervised, pre-training method were used to reduce the data dimensionality. Figure 6 summarizes the findings, which we will go through following. We list each classifier's accuracy, sensitivity, and specificity along with the total time it took to train each model.

## 6 Results and Discussion

In cross-validation folds, the PDNN attained a classification ACC-mean is 0.721% (SENS 75%, SPEC 64%) and a range of 67–72% in individual folds. This is the highest classification to date, according to the literature. The RF classifier achieved a 68% is the ACC-mean (SENS 70%, SPEC 60%), while the SVM classifier achieved a 65% is the ACC-mean (from 62 to 72%, SENS 68%, SPEC 62%). The findings demonstrate DL tactics correctly identified ASD/TC participants in multi-site ABIDE I&II above chance.

Findings demonstrate that algorithm performed better than earlier studies of people with autism spectrum conditions identified using brain activation rs-fMRI ABIDE I&II. The Supplementary material contains results utilizing several brain parcellations. By using autoencoders to make the data less dimensional, the outcomes for SVM classifications remained consistent. On a smaller set of dimensions discovered using autoencoders, we applied SVM without performing any fine-tuning. Lower SVM classification accuracy was obtained after the dimensionality reduction (61% ACC using the first autoencoder with 63% ACC using the first and second autoencoders). For SVM and autoencoder strategy via catalog ASD/TC participants in ABIDE I&II, reduced dimensions may exhibit too complicated patterns. Comparing the deep learning classification approach to SVM, it was found that classification ACC increased by an average of 5%. Additionally, compared to a prior study that sought to categorize ASD using the multi-site ABIDE dataset, the deep learning approach showed a 10% improvement in classification ACC. Comparing the current categorization to studies that sought to classify ASD with smaller participant samples, there was a decrease in specificity and sensitivity. Studies have produced at least 80% classification accuracy levels and as high as 90% (for example) [4, 18, 28]. To evaluate a realistic perspective of how our model would behave in the actual clinical context, we use two metrics, positive and negative prediction values [29]. The computation is based on the correlation between the ACC, SENS, and SPEC of ASD.

The accuracy of classification increases when there are fewer site-wise variances or none at all in the dataset, but accuracy decreases when supervised methods, like ABIDE, are required to classify data across numerous sites. The problem for ML brain imaging investigations is the growth in dimensionality across various datasets. The dimensions might accurately represent variation that contributes therapeutically salutary erudition via vis discerning phrenic snarl, like data from legion stats.

Wilcoxon Signed Ranks Related Groups inquest exclusive categorization technique to analyze the findings further. We specifically evaluated each classification method's label to the actual data. Results demonstrated statistically significant labeling for the SVM classifier ( $Z = 12.08, p = 0.001$ ). The classification accuracy of RF was marginally improved ( $Z = 2.25, p = 0.010$ ); nevertheless, there were still statistically significant differences across labels. When the DNN classifier was applied, there was no statistically significant difference between the labels ( $Z = 0.3$ ,

$p = 0.5$ ). The single classification technique, DNN, did not detect any demographic difference between regiment also actual labels.

Present cramming 72% accuracy boosts the level of expertise. PDNNs enable the learner to express more complex functions, especially when coupled with autoencoders. The literature to date demonstrates that supervised approaches are effective at classifying high-dimensional spaces in less amount of samples. Problems with a huge feature space have their dimensionality successfully reduced by these networks [30]. However, by using intra-site data to train our model using the same hyperparameters and fivefold strategy, the average accuracy was 52%, which was what we achieved. The abundance of data in ABIDE facilitates model generalization, and site diversity prevents overfitting across sites.

We used a cross-validation approach called leave\_one\_site\_out via assess heuristic recital across multi-sites. One site's data was left out of the training procedure, and the model was tested using that site's data as the test set. The goal was to see if the concept could be applied to new, distinct locations. Table 3 presents the findings of these additional analyses. In the supplemental material, Fig. 6. respectively displays the results for SVM, RF, DNN and PDNN. When the accuracy scores were compared to head motion quality metrics, there was no evidence of a head-stirring effect on classification ACC. Capacity of the global blueprint included comp scoop and site-unique variables outwardly sacrificing the SPEC of training data is tested by classifying one site without it.

## 7 Autism Brain Neural Patterns Connectivity

In the ASD rsfMRI data, interaction mid the rs-fMRI data for various brain regions shows twain lucid pat of locations under connected (resentfully intersection) and hugely linked (assuredly interaction): A posterior network of regions whereby activation during rs-fMRI seemed significantly correlated and scattered network involving anterolateral areas in the brain with activation during rs-fMRI seemed negatively associated. An existing idea of anterior–posterior underconnectivity as in autistic brain is investigated along with potential interpretations of these findings. The brain areas with the greatest anti-correlation for people with ASD were the paracingulate cortex, supramarginal parietal, and middle temporal parietal. For our deep learning classification, the anti-correlation patterns of these regions were the most pertinent features. The anti-correlated areas are listed in Tables 4 and 5. Figure 7 displays the regions of the brain for ASD participants that displayed the highest correlation. The Occipital Pole, Lateral Occipital Cortex, Superior Division, and other posterior regions of the brain all had the strongest correlation. After the anti-correlated areas, correlation motif areas prevail the attributes that mattered DL classification.

ASD patients have shown breakdown of the anterior–posterior connection (connection between activation time series) in task-related [5, 6] and rs-fMRI studies [31]. In past research, reduced anterior–posterior connection also increased regional connection in the posterior, relative to the brain connections of reign, observed in the

**Table 4** Leave\_site\_out fivefold cross\_validation outcome employing PDNN-ABIDE-II

Universities	Total number of participants	ACC	SENS	SPEC
<i>Details of ABIDE II dataset</i>				
1_BNI	58	0.68	0.58	0.54
2_EUMCR	54	0.69	0.59	0.56
3_ETHZ	37	0.67	0.66	
4_GU	106	0.66	0.62	0.71
5_IU	40	0.63	0.67	0.4
6_IP&RDH	56	0.67	0.68	0.58
7_KUL	28	0.65	0.69	0.64
8_KKI	211	0.64	0.64	0.63
9_NYULMC_S1	78	0.69	0.65	0.62
10_NYULMC_S2	27	0.7	0.61	0.67
11_ONRC&ILHH	59	0.68	0.66	0.64
12_OH&SU	93	0.69	0.66	0.65
13_TCHS	42	0.69	0.64	0.68
14_SDSU	58	0.68	0.64	0.62
15_SU	42	0.65	0.68	0.58
16_UCD	32	0.64	0.62	0.59
17_UCLA	32	0.63	0.63	0.57
18_UM	28	0.67	0.63	0.6
19_UUSM	33	0.67	0.65	0.61
Mean	58.63	0.667	0.6421	0.5152

**Table 5** Brain region anti-correlated

	Source regions (green)	Red regions	Blue areas
IC0	Paracingulate lobe	Middle temporal lobe; posterior division	Precuneus cortex
IC1	Supramarginal lobe	Inferior frontal lobe	Superior temporal lobe
IC2&3	Middle temporal lobe	Paracingulate lobe	Precuneus cortex, cingulate lobe

brains of autistic patients. An evidence-based explanation of underconnectivity in autism is based on this research and brain imaging data [24]. Additionally, indicators of the anatomy of the brain, notably corpus callosum morphometry, have been linked to the anterior–posterior under the connectivity idea [32].

Previous studies on the brain function of people with ASD suggest that the anterior–posterior brain connection, combined with heightened posterior, or local connectivity, may be impaired in ASD. The current study's results show that the role of more anterior (para-cingulate lobe) and more posterior (supramarginal lobe)

regions, as well as frontal–temporal regions (namely inferior frontal, fusiform lobe, and orbital cortex; middle temporal), were anti-correlated. We propose that under-connectivity between the anterior and posterior ASD brain regions—said to have made the greatest contribution to the current classification—is the root of the anti-correlation. The difference between the two groups was made possible by the trait of anterior–posterior underconnectivity, which also explains how the autistic brain functions.

fMRI is used to track brain activity and how it relates to different brain parts. RS-fMRI is employed via assess regional interactions when a participant is not engaged in an explicit activity. The study of these interactions is crucial to improving our understanding of human behavior, cognitive development, and neuropsychiatric illness because they show network connections between different brain regions. More precisely, this resting brain activity is noninvasively monitored using fMRI, recognizing variations in cerebral blood flow as Blood Oxygen Level Dependent (BOLD) MRI signals.

According to a functional connectivity study based on inter-regional BOLD time series correlation, a specific association results from chance. It may indicate a functional process involving numerous regions or a white matter connection. Discovering a random association requires considering that spontaneous BOLD fluctuations are oscillatory and appear smoothly over space; finding a substantial correlation is, therefore, like finding synchronized waves in a choppy ocean.

Thus, ABIDE I and ABIDE II patients had ADOS data accessible. According to the research, there was no correlation between the networks from Tables 4 and 5 and ADOS score. Such variation introduces noise to the brain imaging data that makes it difficult to infer signatures of brain activation that can categorize disease states; however, the achievement of a reliable classification accuracy despite such noise generated by various equipment and demographics shows promise for machine learning applications to clinical datasets.

## 8 Conclusion

The findings imply DL techniques may accurately identify substantial many sites datasets. Compared to single-site datasets, it is necessary to account for several causes of variance in people, scanning methods, and equipment when classifying across multiple sites. ASD detection of early is very crucial. We have tested many sites data, evaluated different methods with existing models, and implemented with different parameter tuning and training using proposed DNN. Current models are RF, SVM, and DNN with PDNN techniques. The proposed model got 72.10% ACC with two different ABIDE I and II datasets. In the future, we can use hybrid or ensemble deep learning methods for better performance results.

## References

1. Varoquaux G, Thirion B (2014) How machine learning is shaping cognitive neuroimaging. *GigaScience* 3(1):1–7
2. Kassam KS, Markey AR, Cherkassky VL, Loewenstein G, Just MA (2013) Identifying emotions on the basis of neural activation. *PLoS ONE* 8(6):e66032
3. Aylward EH, Minshew NJ, Goldstein G, Honeycutt NA, Augustine AM, Yates KO, Pearlson GD (1999) MRI volumes of amygdala and hippocampus in non–mentally retarded autistic adolescents and adults. *Neurology* 53(9):2145–2145
4. Just MA, Cherkassky VL, Buchweitz A, Keller TA, Mitchell TM (2014) Identifying autism from neural representations of social interactions: neurocognitive markers of autism. *PLoS ONE* 9(12):e113879
5. Kana RK, Keller TA, Cherkassky VL, Minshew NJ, Just MA (2009) Atypical frontal-posterior synchronization of theory of mind regions in autism during mental state attribution. *Soc Neurosci* 4(2):135–152
6. Schipul SE, Williams DL, Keller TA, Minshew NJ, Just MA (2012) Distinctive neural processes during learning in autism. *Cereb Cortex* 22(4):937–950
7. Jordan MI, Mitchell TM (2015) Machine learning: trends, perspectives, and prospects. *Science* 349(6245):255–260
8. Pereira F, Mitchell T, Botvinick M (2009) Machine learning classifiers and fMRI: a tutorial overview. *Neuroimage* 45(1):S199–S209
9. Haxby JV, Gobbini MI, Furey ML, Ishai A, Schouten JL, Pietrini P (2001) Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science* 293(5539):2425–2430
10. O’toole AJ, Jiang F, Abdi H, Haxby JV (2005) Partially distributed representations of objects and faces in ventral temporal cortex. *J Cogn Neurosci* 17(4):580–590
11. Buchweitz A, Shinkareva SV, Mason RA, Mitchell TM, Just MA (2012) Identifying bilingual semantic neural representations across languages. *Brain Lang* 120(3):282–289
12. Mitchell TM, Shinkareva SV, Carlson A, Chang KM, Malave VL, Mason RA, Just MA (2008) Predicting human brain activity associated with the meanings of nouns. *Science* 320(5880):1191–1195
13. Shinkareva SV, Malave VL, Mason RA, Mitchell TM, Just MA (2011) Commonality of neural representations of words and pictures. *Neuroimage* 54(3):2418–2425
14. Bauer AJ, Just MA (2015) Monitoring the growth of the neural representations of new animal concepts. *Hum Brain Mapp* 36(8):3213–3226
15. Yang J, Benyamin B, McEvoy BP, Gordon S, Henders AK, Nyholt DR, Visscher PM (2010) Common SNPs explain a large proportion of the heritability for human height. *Nat Genet* 42(7):565–569
16. Craddock RC, Holtzheimer PE III, Hu XP, Mayberg HS (2009) Disease state prediction from resting state functional connectivity. *Magn Reson Med: An Official J Int Soc Magn Reson Med* 62(6):1619–1628
17. Plitt M, Barnes KA, Martin A (2015) Functional connectivity classification of autism identifies highly predictive brain features but falls short of biomarker standards. *NeuroImage: Clin* 7:359–366
18. Anderson JS, Nielsen JA, Froehlich AL, DuBray MB, Druzgal TJ, Cariello AN, Lainhart JE (2011) Functional connectivity magnetic resonance imaging classification of autism. *Brain* 134(12):3742–3754
19. Abraham A, Milham MP, Di Martino A, Craddock RC, Samaras D, Thirion B, Varoquaux G (2017) Deriving reproducible biomarkers from multi-site resting-state data: an autism-based example. *Neuroimage* 147:736–745
20. Lamani MR, Benadit PJ, Vaithianathan K (2023) Multi-atlas graph convolutional networks and convolutional recurrent neural networks-based ensemble learning for classification of autism spectrum disorders. *SN Computer Science* 4(3):213

21. Di Martino A, Yan CG, Li Q, Denio E, Castellanos FX, Alaerts K, Milham MP (2014) The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Mol Psychiatry* 19(6):659–667
22. Fox MD, Greicius M (2010) Clinical applications of resting state functional connectivity. *Front Syst Neurosci* 4:1443
23. Franco AR, Mannell MV, Calhoun VD, Mayer AR (2013) Impact of analysis methods on the reproducibility and reliability of resting-state networks. *Brain Connectivity* 3(4):363–374
24. Cherkassky VL, Kana RK, Keller TA, Just MA (2006) Functional connectivity in a baseline resting-state network in autism. *NeuroReport* 17(16):1687–1690
25. Biswal B, Zerrin Yetkin F, Haughton VM, Hyde JS (1995) Functional connectivity in the motor cortex of resting human brain using echo-planar MRI. *Magn Reson Med* 34(4):537–541
26. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA, Bottou L (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J Mach Learn Res* 11(12)
27. Vapnik V (1998) The support vector method of function estimation. In: Nonlinear modeling: advanced black-box techniques. Springer Us, Boston, MA, pp 55–85
28. Ho TK (1995) Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition, vol 1. IEEE, pp 278–282
29. Uddin LQ, Supekar K, Menon V (2013) Reconceptualizing functional brain connectivity in autism from a developmental perspective. *Front Hum Neurosci* 7:458
30. Castellanos FX, Di Martino A, Craddock RC, Mehta AD, Milham MP (2013) Clinical applications of the functional connectome. *NeuroImage* 80:527–540
31. Hjelm RD, Calhoun VD, Salakhutdinov R, Allen EA, Adali T, Plis SM (2014) Restricted Boltzmann machines for neuroimaging: an application in identifying intrinsic networks. *NeuroImage* 96:245–260
32. Just MA, Cherkassky VL, Keller TA, Minshew NJ (2004) Cortical activation and synchronization during sentence comprehension in high-functioning autism: evidence of underconnectivity. *Brain* 127(8):1811–1821

# Comparison of Machine Learning-Based Intrusion Detection Systems Using UNSW-NB15 Dataset



Rakoth Kandan Sambandam, D. Daniel, R. Gokulapriya, Divya Vetriveeran, J. Jenefa, and Anuneshwar

**Abstract** Various machine learning classifiers have been employed recently to enhance network intrusion detection. In the literature, researchers have put forth a wide range of intrusion detection solutions. The accuracy of the machine learning classifiers' intrusion detection is limited by the fact that they were trained on dated samples. Therefore, the most recent dataset must be used to train the machine learning classifiers. In this study, UNSW-NB15, machine learning classifiers are trained using the most recent dataset. A taxonomy of classifiers based on eager and lazy learners is used to train the chosen classifiers, such as K-Means (KNN), Polynomial Features, Random Forest (RF), and Naive Bayes (NB), Linear Regression. In order to decrease the redundant and unnecessary features in the UNSW-NB15 dataset, chi-Square, a filter-based feature selection technique, is used in this study. When comparing these machine learning classifiers, performance is measured in terms of accuracy, mean squared error (MSE), precision, recall, and F1-score with or without feature selection technique.

**Keywords** Intrusion detection · UNSW-NB15 · Cyber security · Machine learning

## 1 Introduction

Cybersecurity is primarily about integrating people, processes, and technologies to cover the full spectrum of threat reduction, vulnerability reduction, deterrence, international engagement, incident response, resiliency, and recovery policies and activities, including computer network operations, information assurance, law enforcement, etc. or in other words networks, computers, programs, and data are all protected by a group of technologies, procedures, and practices known as cyber security [1] from attack, unauthorized access, and damage. Data that is stored, communicated, or used on an information system is referred to as “digital data” when discussing cyber

---

R. K. Sambandam (✉) · D. Daniel · R. Gokulapriya · D. Vetriveeran · J. Jenefa · Anuneshwar  
Department of CSE, SOET, CHRIST (Deemed to be University), Kengeri Campus, Bangalore,  
India  
e-mail: [rakothsen@gmail.com](mailto:rakothsen@gmail.com)

security strategies and practices [2]. Security relates to the protection, which includes the security of systems, networks, applications, and information, whereas cyber is related to technology that incorporates systems, networks, programs, or data [3].

## 2 Security Policies

Security policies [4] are rules and guidelines implemented by organizations to ensure the confidentiality, integrity, and availability of their information and resources [5]. They define the purpose, scope, roles, and responsibilities of individuals and departments involved in the security program. Access control policies determine how access to information is granted and revoked. Incident response policies outline procedures for handling security incidents. Data protection policies safeguard sensitive data through classification, storage, and transmission guidelines. Network security policies protect network infrastructure through measures like segmentation, access control, and intrusion detection. Regular policy review and updates are crucial to stay aligned with evolving security threats and best practices.

## 3 Dataset Description

The UNSW-NB-15 dataset is a network intrusion detection dataset created by the University of New South Wales. It contains 2.5 million network packets captured from a virtual network environment. Various network applications and services were run to generate the traffic, including email, web browsing, file transfer, and remote login. The traffic was then subjected to different types of attacks such as denial-of-service (DoS), port scanning, and remote-to-local (R2L) attacks. The dataset is divided into a training set (approximately 1.2 million packets) and a testing set (approximately 1.3 million packets). It includes five classes of attacks: normal traffic, DoS, probing, R2L, and U2R (user-to-root) attacks. The features are categorized into basic, content, traffic, time-related, connection-related, and host-based features.

### 3.1 Objectives

The aim of this work is to build Attack Prediction which analyzes the given input and displays an output as whether an attack is present or not. The main objectives are

- Gaining knowledge about different types of attacks
- Learning various machine learning techniques
- To develop a model which provides better results.

## 4 Literature Review

There have been many research in the field, the most common algorithms were reviewed, and it is found that random forest performs better. An overall comparison of the works is given in Table 1.

### 4.1 *Inferences Drawn from Literature Review*

The two-stage classification method for NIDS utilizing the UNSW-NB15 dataset has shown to be successful in enhancing the performance of attack-category classifiers; the study also identified a number of drawbacks, and the restriction is made even more evident by the uneven distribution of data entries for each type of assault in the training and testing sets. FARs, however, did not always increase or decrease in response to feature selection—in some cases, they even remained the same. Nevertheless, overall, the FARs for the rare attacks were low and outperformed the NB algorithm with feature selection.

## 5 Problem Formulation and Proposed Work

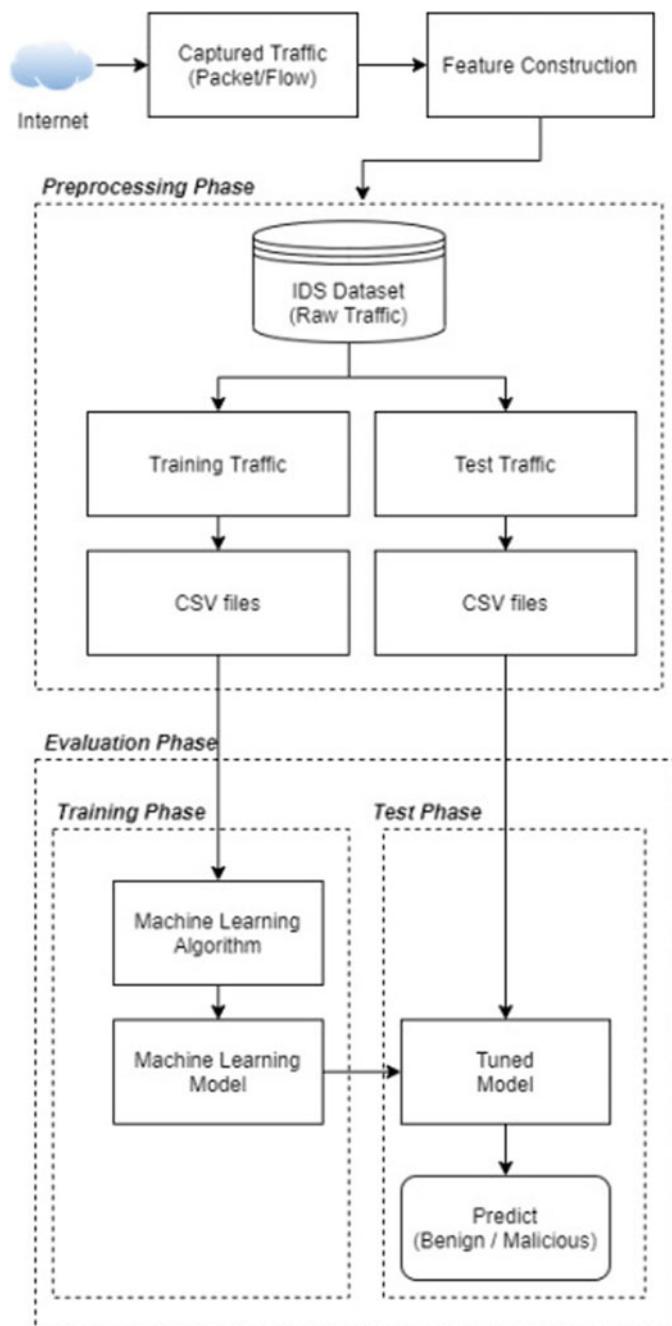
The growth in network traffic has contributed to the growth in data and network security breaches. Network intrusions or attacks are events transmitted through network packets that can compromise the confidentiality, integrity, and availability of computer systems [14]. In this study, data is classified using Naive Bayes, k-means, linear regression, and polynomial regression algorithms [15] and evaluated using a correlation-based subset for both binary and multiclass classifications. The overall representation is given in Fig. 1.

### Proposed Work

The research proposes a system with an architectural design for data processing in machine learning. It emphasizes the importance of data architecture in ensuring successful learning. The data processing phase includes cleansing, normalization, and feature selection using a filter-based approach. The selected features are used to train and validate machine learning classifiers: LR, NB, RF, LSTM, K-Means, and polynomial regression. The experiments are conducted using Python on specific hardware. The dataset is preprocessed, split into training and testing sets, and the classifiers are trained and evaluated based on various criteria. The RF classifier achieves the highest accuracy (97.8%), while the K-means classifier with cluster = 4 exhibits the highest mean squared error and lowest accuracy among the chosen classifiers.

**Table 1** Comparison of existing works

Paper title	ML	Application	Results
Comparing Machine Learning Techniques for Intrusion Detection Using UNSW-NB15 Dataset (2018) [6]	Decision trees, random forests, support vector machines	Network intrusion detection	Best performance achieved with random forest classifier
Anomaly-Based Intrusion Detection for IoT Using Machine Learning Techniques on UNSW-NB15 Dataset (2019) [7]	Decision trees, artificial neural networks	IoT intrusion detection	High accuracy, low false positive rates
Machine Learning-Based Intrusion Detection System for Wireless Sensor Networks Using UNSW-NB15 Dataset (2020) [8]	Decision trees, random forests	Wireless sensor network intrusion detection	High detection rates, low false positive rates
A Novel Hybrid Intrusion Detection System for Industrial Control Networks Using UNSW-NB15 Dataset (2019) [9]	Support vector machines, k-nearest neighbors, rule-based expert system	Industrial control network intrusion detection	High detection rates, low false positive rates
Comparison of Machine Learning Models for Network Intrusion Detection Using UNSW-NB15 Dataset (2019) [10]	Decision trees, random forests, neural networks	Network intrusion detection	Best performance achieved with random forest classifier
Evaluation of Machine Learning Techniques for Network Intrusion Detection on UNSW-NB15 Dataset (2019) [11]	Decision trees, support vector machines, naive Bayes	Network intrusion detection	High detection rates, low false positive rates for several techniques
Comparison of Machine Learning Algorithms for Intrusion Detection in IoT Networks (2020) [12]	Decision trees, support vector machines, artificial neural networks	IoT intrusion detection	Best performance achieved with random forest algorithm
FI-PCA for IoT Network Intrusion Detection (2020) [13]	Machine learning and neural networks comparison	IoT intrusion detection	Framework to support several ML algorithms



**Fig. 1** Proposed work

## 5.1 *Implementation*

The preprocessing procedures are carried out and then the methodology is applied. The null values that are present in the dataset are first addressed during preprocessing. Utilizing a label encoder, category data is transformed into numerical form. The relationship between the values obtained from label encoding is then broken using one hot encoder. The preprocessed data is then divided into training and testing groups. The models are built using K-means, LR, NB, PL, DT, and RF classifiers. Then, these models are used to predict the labels of test data. Between actual labels and expected labels, a comparison is made. Accuracy, precision, mean square error, recall, and F1-score are performance indicators used to assess the models.

## 6 Results and Discussions

The results are analyzed as follows for both the binary and multiclass classification. The binary classifier performances are compared as given in the following section.

### 6.1 *Linear Regression*

Mean Absolute Error—0.24077610101632277

Mean Squared Error—0.24077610101632277

Root Mean Squared Error—0.4906894140047478

Accuracy—75.92238989836773.

The model's performance is poor, with an accuracy score of 0.76. Precision is 0.76 for the abnormal class and 0 for the normal class. Recall is 1.00 for the abnormal class and 0 for the normal class. The F1-score is 0.86 for the abnormal class but 0 for the normal class. This could be due to class imbalance, feature selection, or model selection.

## 6.2 Logistic Regression

	Precision	Recall	F1-score	Support
Abnormal	0.97	1.00	0.99	12,326
Normal	0.99	0.91	0.95	3909
Accuracy			0.98	16,235

The model performs very well with high accuracy, precision, recall, and F1-scores for both the abnormal and normal classes. These high scores indicate that the model effectively captures patterns in the data and makes accurate predictions.

## 6.3 Linear SVM

	Precision	Recall	F1-score	Support
Abnormal	0.97	1.00	0.99	12,326
Normal	1.00	0.91	0.95	3909
Accuracy			0.98	16,235

The model achieves a high accuracy of 0.98, indicating accurate predictions. It has high precision and recall scores for both classes, reflecting low false positive rates and the ability to identify most instances. The F1-score is also high for both classes, indicating overall strong performance.

## 6.4 KNN

	Precision	Recall	F1-score	Support
Abnormal	0.99	0.99	0.99	12,326
Normal	0.97	0.96	0.96	3909
Accuracy			0.98	16,235

The model achieves a high accuracy of 0.98, indicating accurate predictions. It has high precision and recall scores for both classes, suggesting minimal false positive predictions and effective identification of instances. The F1-scores are also high for both classes, reflecting strong overall performance.

## 6.5 Random Forest

	Precision	Recall	F1-score	Support
Abnormal	0.99	0.99	0.99	12,326
Normal	0.98	0.96	0.97	3909
Accuracy			0.99	16,235

The model demonstrates excellent binary classification performance, with high precision, recall, and F1-scores for both classes. It makes very few false positive errors and correctly identifies most instances. However, the recall for the “normal” class is slightly lower, suggesting some misclassification. Overall, the model achieves an impressive accuracy of 0.99, demonstrating accurate predictions for the majority of instances.

## 6.6 Decision Tree

	Precision	Recall	F1-score	Support
Abnormal	0.99	0.99	0.99	12,326
Normal	0.96	0.97	0.96	3909
Accuracy			0.98	16,235

The model performs well with an overall accuracy of 0.98. The precision and recall values are high for both classes, indicating that the model correctly identifies most instances. The precision, recall, and F1-score are all high for the “abnormal” class, indicating accurate identification of “abnormal” instances. The same is true for the “normal” class, although the precision is slightly lower. Overall, the model has performed well and can be considered a good fit for the classification problem.

## 6.7 Multilayer Perceptron

	Precision	Recall	F1-score	Support
Abnormal	0.98	0.99	0.99	12,326
Normal	0.98	0.95	0.97	3909
Accuracy			0.98	16,235
Macro avg	0.98	0.97	0.98	16,235
Weighted avg	0.98	0.98	0.98	16,235

The model has high-precision values for both classes, indicating accurate predictions. It has higher recall for abnormal samples but struggles slightly with normal samples. The F1-score demonstrates a good balance between precision and recall. The model performs well in identifying abnormal samples but has room for improvement in identifying normal samples.

## 6.8 Multiclass Classifiers

### a. Linear Regression

	Precision	Recall	F1-score	Support
Analysis	0.00	0.00	0.00	166
Backdoor	0.00	1.00	0.00	32
DoS	0.00	0.00	0.00	521
Exploits	0.00	0.00	0.00	4900
Fuzzers	0.00	0.00	0.00	508
Generic	0.00	0.00	0.00	11,839
Normal	0.00	0.00	0.00	5855
Reconnaissance	0.00	0.00	0.00	502
Worms	0.00	0.00	0.00	29
Accuracy			0.00	24,352

The model's performance on the classification problem is poor. The precision, recall, and F1-score values for most of the classes are zero, indicating that the model fails to correctly classify instances into those classes. The overall accuracy and F1-score are also low, suggesting that the model struggles to classify instances correctly on average. Significant improvements are required for the model to effectively address this classification problem.

### b. Logistic Regression

	Precision	Recall	F1-score	Support
Analysis	1.00	1.00	1.00	166
Backdoor	0.00	0.00	0.00	32
DoS	1.00	1.00	1.00	521
Exploits	1.00	1.00	1.00	4900
Fuzzers	0.56	0.42	0.48	508
Generic	0.99	0.99	0.99	11,839
Normal	1.00	1.00	1.00	5855
Reconnaissance	0.54	0.76	0.63	502
Worms	0.00	0.00	0.00	29
Accuracy			0.98	24,352

The classification report shows varying performance for different classes. The model accurately predicts some classes but struggles with others. Overall, the model's performance is mediocre, with an accuracy of 0.98 and a macro avg F1-score of 0.68. Enhancements are needed to improve predictions for the underperforming classes.

### c. Linear SVM

	Precision	Recall	F1-score	Support
Analysis	1.00	1.00	1.00	166
Backdoor	0.00	0.00	0.00	32
DoS	1.00	1.00	1.00	521
Exploits	1.00	1.00	1.00	4900
Fuzzers	0.54	0.47	0.50	508
Generic	0.99	0.99	0.99	11,839
Normal	1.00	1.00	1.00	5855
Reconnaissance	0.56	0.71	0.62	502
Worms	0.00	0.00	0.00	29
Accuracy	0.98	24,352		

The classification report shows that the model performs well for some classes, with perfect precision, recall, and F1-score values. However, it struggles with other classes, where the metrics are 0.00, indicating no correct predictions. The model achieves moderate performance for some classes, with precision, recall, and F1-score values ranging from 0.50 to 0.71. The weighted average metrics indicate overall good performance, but certain classes need improvement.

#### d. KNN Classifier

	Precision	Recall	F1-score	Support
Analysis	1.00	1.00	1.00	166
Backdoor	0.00	0.00	0.00	32
DoS	1.00	1.00	1.00	521
Exploits	1.00	1.00	1.00	4900
Fuzzers	0.48	0.52	0.50	508
Generic	0.99	0.99	0.99	11,839
Normal	1.00	1.00	1.00	5855
Reconnaissance	0.55	0.54	0.55	502
Worms	0.00	0.00	0.00	29
Accuracy			0.97	24,352

The model consistently performs well for some classes, with precision, recall, and F1-scores close to 1.0. However, its performance is poorer for other classes, with lower precision, recall, and F1-scores. The macro-averaged F1-score indicates that the overall performance across all classes is not satisfactory, but the weighted-averaged F1-score suggests that the model performs well for the majority of the data. Further improvement is needed, particularly for the challenging classes.

#### e. Random Forest Classifier

	Precision	Recall	F1-score	Support
Analysis	1.00	1.00	1.00	166
Backdoor	0.08	0.03	0.05	32
DoS	1.00	1.00	1.00	521
Exploits	1.00	1.00	1.00	4900
Fuzzers	0.49	0.43	0.46	508
Generic	0.99	0.99	0.99	11,839
Normal	1.00	1.00	1.00	5855
Reconnaissance	0.54	0.60	0.57	502
Worms	0.12	0.07	0.09	29
Accuracy			0.97	24,352

The report shows high scores for some classes, indicating accurate predictions. However, other classes have low scores, suggesting difficulty in distinguishing them. Some classes have moderate performance. The weighted average F1-score is 0.97, indicating overall high accuracy. It's important to consider other factors, such as class imbalance, when assessing the model's performance.

#### f. Decision Tree

	Precision	Recall	F1-score	Support
Analysis	1.00	1.00	1.00	166
Backdoor	0.08	0.06	0.07	32
DoS	1.00	1.00	1.00	521
Exploits	1.00	1.00	1.00	4900
Fuzzers	0.50	0.39	0.44	508
Generic	0.98	0.99	0.99	11,839
Normal	1.00	1.00	1.00	5855
Reconnaissance	0.54	0.56	0.55	502
Worms	0.05	0.07	0.06	29
Accuracy			0.97	24,352

The classification report evaluates a machine learning model's performance on a multiclass problem. The model performs well for some classes with high precision, recall, and F1-scores. However, it struggles with other classes, showing lower performance. The macro average F1-score is 0.68, indicating overall moderate performance across all classes. The weighted average F1-score is 0.97, reflecting high accuracy when accounting for class imbalances.

#### g. Multilayer Perceptron

The model's performance in this classification report is consistent with the previous reports. It achieves high precision, recall, and F1-scores for some classes, while struggling with others. The F1-score for one class has slightly decreased, indicating misclassifications. The macro average F1-score remains moderate, and the weighted average F1-score is high, indicating overall high accuracy. Further improvements are needed for consistent and accurate performance across all classes. A comparison of all the algorithms of a binary classification is tabulated below as in Table 2. As shown, it can be seen that the random forest algorithm performed better in terms of a binary classification. A comparison of all the algorithms of a multiclass classification is tabulated below as in Table 3. Based on the experiments, the linear SVM performs better with the least mean error.

## 7 Conclusion and Future Scope

This paper had explored the various algorithms for binary as well as multiclass classification. Among the results, it is found that the multilayer perceptron performs well for both the classification. Deep learning is an improvement to the neural network.

**Table 2** Binary classification performance analysis

Algorithms	Accuracy	R2 score	Mean absolute error	Mean squared error
Linear Regression	96.85	92.02	0.04	0.09
Logistic Regression	97.65	86.36	0.05	0.157
K-Means	1.46	- 63.34	3.06	10.76
Long Short-Time Memory	97.659	86.17	0.05	0.159
Random Forest	97.82	88.69	0.04	0.13
Decision Tree	97.66	87.89	0.05	0.13
Polynomial Regression	92.95	92.3	0.08	0.08

**Table 3** Multiclass classification performance analysis

Algorithms	Mean absolute error	Mean absolute error	Accuracy
Linear Regression	3.77	3.95	0.13
Logistic Regression	0.06	0.42	97.5
Linear Support Vector Machine	0.05	0.42	97.5
K-Nearest Neighbor Classifier	0.06	0.44	97.3
Random Forest	0.06	0.44	97.3
Decision Tree	0.068	0.45	97.19
Multilayer Perceptron	0.06	0.42	97.5

It has gained popularity in recent years. Using this new method, the present IDS can be enhanced. Deep learning methods are categorized according to their architecture. There are three sorts of algorithms: generative (unsupervised), discriminative (supervised), and hybrid. To enhance efficiency and reduce training time, we require powerful computer resources, which are both expensive and scarce. Reinforcement learning (RL) is a new subject in which research is being conducted. Deep reinforcement learning may also be used as the next stage in intrusion detection. Future scopes are offered to assist researchers in developing more effective methods of detecting assaults. There is a description of existing literature that uses procedures comparable to the UNSW NB-15 dataset.

**Acknowledgements** The authors gratefully acknowledge the authorities of CHRIST (Deemed to be University), Bengaluru, Karnataka, for the facilities offered to carry out this work.

## References

1. Sağlam RB, Miller V, Franqueira VNL (2023) A systematic literature review on cyber security education for children. *IEEE Trans Educ* 66(3):274–286
2. Gaspar J, Cruz T, Lam C-T, Simões P, Smart substation communications and cybersecurity: a comprehensive survey. In: *IEEE Commun Surv Tutor*
3. Arshad I, Alsamhi SH, Qiao Y, Lee B, Ye Y (2023) A novel framework for smart cyber defence: a deep-dive into deep learning attacks and defences. *IEEE Access* 11:88527–88548
4. Alassaf M, Alkhailifah A (2021) Exploring the influence of direct and indirect factors on information security policy compliance: a systematic literature review. *IEEE Access* 9:162687–162705
5. Fan J et al (2023) Understanding security in smart city domains from the ANT-centric perspective. In: *IEEE Internet of Things J* 10(13):11199–11223
6. Moustafa K, Slay J, Creech A (2018) Comparing machine learning techniques for intrusion detection using UNSW-NB15 dataset. In: 2018 17th IEEE International conference on machine learning and applications (ICMLA), Dec 2018, pp 1117–1122. <https://doi.org/10.1109/ICMLA.2018.00166>
7. Moustafa K, Slay J, Creech A (2019) Anomaly-based intrusion detection for IoT using machine learning techniques on UNSW-NB15 dataset. In: 2019 18th IEEE International conference on machine learning and applications (ICMLA), Dec 2019, pp 905–912. <https://doi.org/10.1109/ICMLA.2019.00153>
8. Moustafa K, Slay S, Creech A (2020) Machine learning-based intrusion detection system for wireless sensor networks using UNSW-NB15 dataset. In: 2020 19th IEEE International conference on machine learning and applications (ICMLA), Dec 2020, pp 1084–1089. <https://doi.org/10.1109/ICMLA51294.2020.00170>
9. Al-Dmour A, Al-Shawabkeh M, Al-Khasawneh S (2019) A novel hybrid intrusion detection system for industrial control networks using UNSW-NB15 dataset. In: 2019 International conference on computer and applications (ICCA), Nov 2019, pp 148–152. <https://doi.org/10.1109/COMAPP.2019.8924676>
10. Kumar N, Goyal A, Agrawal RK (2019) Comparison of machine learning models for network intrusion detection using UNSW-NB15 dataset. In: 2019 2nd International conference on innovations in electronics, signal processing and communication (IESC), Mar 2019, pp 237–240. [https://doi.org/10.1109/IESC\\_2019.8724416](https://doi.org/10.1109/IESC_2019.8724416)
11. Hussain MS, Faisal MS, Ali A (2019) Evaluation of machine learning techniques for network intrusion detection on UNSW-NB15 dataset. In: 2019 IEEE International symposium on signal processing and information technology (ISSPIT), Dec 2019, pp 78–83. <https://doi.org/10.1109/ISSPIT46623.2019.8985941>
12. Ali KM, Ullah FZ (2020) Comparison of machine learning algorithms for intrusion detection in IoT networks. In: 2020 IEEE 7th international conference on industrial engineering and applications (ICIEA), Aug 2020, pp 1–6. <https://doi.org/10.1109/ICIEA48952.2020.9144148>
13. Abdulkareem SA, Foh CH, Carrez F, Moessner K (2022) FI-PCA for IoT network intrusion detection. In: 2022 International symposium on networks, computers and communications (ISNCC). IEEE, pp 1–6
14. Guo F (2023) Research on digital image information security and encryption technology. In: 2023 International conference on computer graphics and image processing (CGIP), Tokyo, Japan, 2023, pp 78–81
15. Dixit U, Bhatia S, Bhatia P (2022) Comparison of different machine learning algorithms based on intrusion detection system. In: 2022 International conference on machine learning, Big Data, cloud and parallel computing (COM-IT-CON), Faridabad, India, pp 667–672

# Comparative Analysis of Face Recognition Models for Criminal Detection



Gouri Goyal, Vaibhav Kumar, Piyush Aggarwal, Gunjan Chugh,  
and Tripti Lamba

**Abstract** Facial recognition technology has gained significant attention in recent years for its potential in improving criminal detection and enhancing security systems. This paper compares and contrasts different facial recognition technologies which are being implemented in video datasets for criminal detection. Law enforcement organizations are vigilantly monitoring facial recognition technologies as an approach to finding and apprehending criminals. The objective is to evaluate and compare how different deep learning techniques perform in real criminal detecting circumstances. A comprehensive video dataset that includes footage from law enforcement archives and surveillance camera footage is curated. To ensure a representative sample for the study, the dataset consists of a group of people with an assortment of physical attributes and notoriety. Each facial recognition technology is used and refined through transfer learning on the curated video dataset. To examine the performance of the systems, accuracy, precision, and error rate are employed as performance assessment variables. The comparative analysis highlights both the advantages and disadvantages of each approach. VGGFace, FaceNet, OpenFace, and DeepFace models are being assessed where the VGGFace model depicted the highest accuracy. The findings offer valuable insights into the performance and applications of various facial recognition systems. The findings expand facial recognition technology and demonstrate the value of law enforcement tools in strengthening public safety.

**Keywords** Deep learning · VGGFace · FaceNet · OpenFace · DeepFace · Facial recognition · Criminal identification

---

G. Goyal (✉) · V. Kumar · P. Aggarwal · G. Chugh · T. Lamba

Department of Artificial Intelligence and Machine Learning, Maharaja Agrasen Institute of Technology, New Delhi, India

e-mail: [gourigoyal2506@gmail.com](mailto:gourigoyal2506@gmail.com)

G. Chugh

e-mail: [gunjanchugh@mait.ac.in](mailto:gunjanchugh@mait.ac.in)

T. Lamba

e-mail: [triptilamba@mait.ac.in](mailto:triptilamba@mait.ac.in)

## 1 Introduction

Facial recognition systems have emerged as a powerful tool for criminal detection and have gained significant attention in law enforcement and security domains. It is becoming one of the most significant applications for commercials and law enforcement which include forensic identification, access controls, and border surveillance [1, 2]. The ability to precisely recognize people from facial photographs or videos has created new opportunities for increasing security precautions, assisting investigations, and enhancing public safety. Deep learning techniques have recently revolutionized the field of facial recognition by producing extremely reliable and accurate models for identifying people in a wide range of circumstances [3]. Currently, face recognition is used as a technology to provide multiple security in various practices like verification of identity, access authority, and observation, to replace passwords and identity cards that are no longer safe [4]. Face recognition involves the following steps: (i) face detection, (ii) face alignment, (iii) numerical representation, and (iv) face recognition can be of two types: (i) face verification and (ii) face identification [5]. The research presented here compares and contrasts the four most extensively used facial recognition methods: VGGFace [6], FaceNet [7], OpenFace [8], and DeepFace [9]. These methods have been specifically used in criminal detection scenarios and have demonstrated good results in a variety of computer vision applications. This study seeks to provide insights into the strengths and limits of each technique, enabling researchers and practitioners in selecting the most appropriate approach for criminal detection applications by comparing their performance using video datasets. A selected video dataset will be used to compare various methods in the comparative study. This study compares VGGFace, FaceNet, OpenFace, and DeepFace to highlight their accuracy, robustness, computing efficiency, and applicability for criminal identification utilizing video datasets. The findings of this study will help the development of facial recognition technologies for criminal detection. Law enforcement agencies will derive advantages greatly from the findings, which will assist them identify the best facial recognition method for duties involving criminal identification.

## 2 Literature Review

The application of transfer learning and deep learning models, such as VGG, FaceNet, DeepFace, and OpenFace, in the field of criminal detection has been widely explored, leading to significant advancements in this domain. Several studies have focused on leveraging these models for accurate and efficient criminal identification and detection. U. Scherhag, R. Raghavendra, K. B. Raja, M. Gomez-Barrero, C. Rathgeb, and C. Busch investigated face recognition systems' vulnerability to morphing face attacks [6]. Their research sought to investigate the effect of altered faces on various face recognition algorithms. The authors shed light on the potential flaws

of facial recognition systems as well as the importance of resilience against such attacks. To achieve pose-invariant face recognition, Mingjie He et al. suggested a deformable convolutional neural network termed “Deformable FaceNet” [7]. Their model was designed to deal with position variations and increase facial recognition performance. The authors used deformable convolutional layers to improve the network’s ability to record and align facial characteristics, resulting in improved pose robustness. OpenFace was created by Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan as a general-purpose facial recognition framework with mobile applications [8]. OpenFace is a facial recognition software that focuses on efficiency and usability. The library provides a variety of functions for face recognition tasks, making it suitable for use in a variety of applications and platforms. Mei Wang and Weihong Deng conducted a thorough investigation of deep facial recognition algorithms [9]. Their survey study discusses network topologies, loss functions, face alignment, and training methodologies, among other topics, in-depth. The authors discuss accomplishments, problems, and prospects in deep facial recognition. These related works collectively highlight the effectiveness and applicability of transfer learning models, including VGG, FaceNet, OpenFace, and DeepFace, in criminal detection scenarios. They demonstrate the importance of comprehensive datasets, fine-tuning techniques, and the extraction of discriminative facial features for achieving accurate and efficient criminal identification and detection.

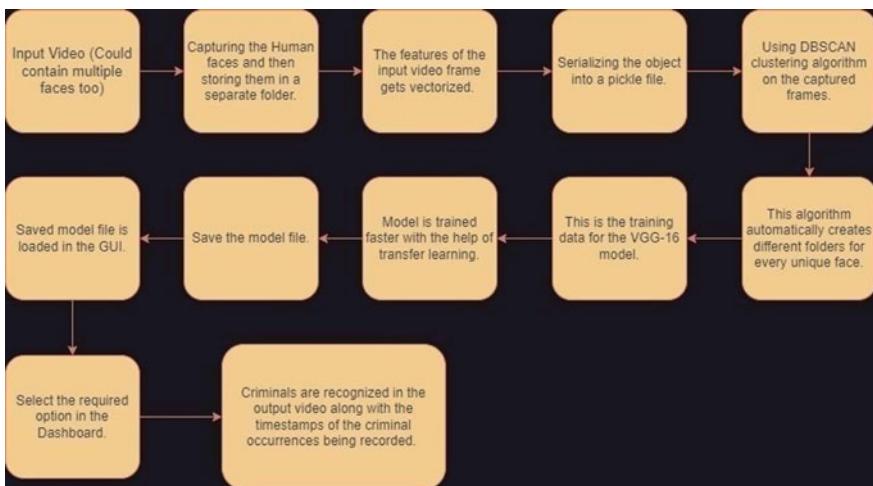
### 3 Methodology

#### 3.1 Data Collection and Preprocessing

Using the VGGFace model and OpenCV’s Haar cascade classifier, the code finds and recognizes faces. The code imports the required libraries, which include OpenCV, Keras, and NumPy. Keras’ Sequential API is used to define the VGGFace model. This is a deep convolutional neural network that has been trained on a big dataset of facial photographs. The VGGFace model’s weights file (`vgg_face_weights.h5`) was obtained by: <https://www.kaggle.com/datasets/acharyarupak391/vggfaceweights>. The Haar cascade classifier (`haarcascade_frontalface_default.xml`) file is in the same directory as the script. A database directory contains photographs of well-known faces named “`name.jpg`”. When you run the code, a window with a video file will appear. If the detected faces match the known faces in the database directory, they will be labeled with their matching names.

### 3.2 Architecture

The systematic methodology presented in this research is intended to improve facial recognition and criminal detection through a thorough integration of video analysis and transfer learning. Human facial extraction is started with video inputs that last between 30 and 60 s, and then these faces are assembled into a single repository of clipped frames. After that, feature representations are created by vectorizing the detailed characteristics of the video frames. Into a pickle object, these representations are serialized. The pickle object, which contains the feature vectors of all the extracted faces, is subjected to the DBSCAN clustering method, which groups the frames into discrete clusters. The dataset organization is optimized by this clustering technique, which creates customized folders for each distinct face. The subsequent step uses these distinct facial datasets to train a VGG-16 model, with accelerated learning made possible by transfer learning strategies. When the trained model reaches the necessary degree of skill, it is persistent, allowing for a smooth integration into a GUI application. This user-friendly interface enables real-time detection of people in output videos by empowering users to select operational parameters via an easy dashboard. The framework, which is significant, allows for situations in which test or validation movies include many faces, highlighting its usefulness and adaptability in complex situations. Through the fusion of cutting-edge approaches, this research offers a systematic way to improve facial recognition and criminal detection as mentioned in flowchart below shown in Fig. 1.



**Fig. 1** System architecture

## 4 Models

### 4.1 VGGFace [6]

VGGFace is a deep learning model designed for face recognition tasks. The VGGNet architecture, which was developed initially for image classification, serves as its foundation. The VGGFace model was created with the specific purpose of extracting facial features from pictures and performing face recognition tasks. Convolutional layers with tiny filter sizes ( $3 \times 3$ ) and layers with maximum pooling make up the VGGFace model. It is a deep model with three fully connected layers after a total of 16 convolutional layers. In contrast to prior models like AlexNet, the model architecture is wider and deeper, which enables it to learn patterns and features that are more complex. The VGGFace model was trained using a sizable face dataset. The last layer is the classifier which is a softmax layer to classify an image to which the individual face class belongs. [4] As mentioned in Table 1, the VGGFace architecture, spanning 19 layers, integrates convolution and pooling operations for hierarchical feature extraction. Dense layers culminate in classification, including an embedding layer for recognition. Its effectiveness stems from a significant parameter count.

**Table 1** VGGFace model architecture

Layer	Output shape	Number of parameters
Input	(Batch,224,224,3)	0
Conv3-64	(Batch,224,224,64)	38,270
MaxPooling	(Batch,112,112,64)	0
Conv3-128	(Batch,112,112,128)	221,440
MaxPooling	(Batch,56,56,128)	0
Conv3-256	(Batch,56,56,256)	885,248
Conv3-256	(Batch,28,28,256)	1,770,496
MaxPooling	(Batch,28,28,256)	0
Conv3-512	(Batch,28,28,512)	3,540,992
Conv3-512	(Batch,28,28,512)	3,540,992
Conv3-512	(Batch,28,28,512)	3,540,992
MaxPooling	(Batch,14,14,512)	0
Conv3-512	(Batch,14,14,512)	3,540,992
Conv3-512	(Batch,14,14,512)	3,540,992
Conv3-512	(Batch,14,14,512)	3,540,992
MaxPooling	(Batch,7,7,512)	0
Flatten	(Batch,25,088)	0
Dense	(Batch,4096)	123,802,624
Dense (Embedding)	(Batch,2622)	10,758,234

**Table 2** FaceNet model architecture

Layer	Output shape	Number of parameters
Input	(Batch,160,160,3)	0
Conv3-64	(Batch,40,40,64)	1792
MaxPooling2D	(Batch,20,20,64)	0
Conv3-128	(Batch,10,10,128)	73,856
Conv3-256	(Batch,5,5,256)	590,080
Conv3-512	(Batch,2,2,512)	2,359,808
Flatten	(Batch,2048)	0
Dense	(Batch,128)	262,272
L2 Normalization	(Batch,128)	0

## 4.2 FaceNet [7]

FaceNet is a deep learning model for face recognition and verification created by Google's machine learning team. It is made to be trained on facial photographs to develop a high-dimensional representation known as face embedding [10]. To extract information from facial photographs, the FaceNet model uses a deep convolutional neural network (CNN) architecture. With the use of a Siamese network structure, it learns to map pairs of face photographs into a common feature space by sharing weights between two identical CNNs [11]. This structure allows the model to develop comparable embedding for photographs of the same person while developing different embedding for images of other people. By minimizing the distance between embedding of the same identity and maximizing the distance between embedding of different identities, the model learns to optimize the embedding. FaceNet is trained using between 100 and 200 M training face thumbnails consisting of about 8 M different identities with input sizes ranging from  $96 \times 96$  pixels to  $224 \times 224$  pixels. As mentioned in Table 2, the FaceNet model, comprising 9 layers, integrates convolutions, pooling, and dense operations. It progressively extracts features, achieving efficient embedding for face recognition. Model complexity arises from substantial parameter counts.

## 4.3 OpenFace [8]

OpenFace is a face recognition and facial landmark detection deep learning model. It was developed to serve as a simple, open-source platform for face analysis jobs. Convolutional neural networks (CNNs), which are the foundation of OpenFace, are used to extract facial features. A sizable dataset of labeled faces, such as the Labeled Faces in the Wild (LFW) dataset, is used to train the OpenFace model [12, 13]. The model gains the ability to separate distinguishing features from faces during training and map those features into a high-dimensional feature space, where the faces of the

**Table 3** OpenFace model architecture

Layer	Operation	Output shape	Number of parameters
Input	–	(Batch,96,96,3)	0
Conv3-64	Convolutional	(Batch,48,48,64)	1792
MaxPooling2D	Maxpooling	(Batch,24,24,64)	0
Conv3-64	Convolutional	(Batch,24,24,64)	73,856
Conv3-128	Convolutional	(Batch,20,20,128)	295,168
Conv3-128	Convolutional	(Batch,18,18,128)	590,080
Conv3-256	Convolutional	(Batch,16,16,256)	1,180,160
Conv3-256	Convolutional	(Batch,14,14,256)	2,359,808
Conv3-512	Convolutional	(Batch,12,12,512)	2,359,808
Conv3-512	Convolutional	(Batch,10,10,512)	2,359,808
Conv3-512	Convolutional	(Batch,8,8,512)	–
Conv3-512	Convolutional	(Batch,6,6,512)	–
Flatten	–	(Batch,18,432)	18,874,368
Dense	–	(Batch,128)	0

same person are located closer together and those of different persons are located farther away. The OpenFace model can be applied to a variety of face analysis tasks after training. It can extract facial features from an input image, including high-dimensional face embedding and facial landmarks [14]. As mentioned in Table 3, the OpenFace model, encompassing 14 layers, implements convolutional and pooling operations for incremental feature extraction. Dense layers and flattening culminate in effective facial embedding. Model complexity arises from substantial parameters.

#### 4.4 DeepFace [7]

DeepFace is a deep learning model for facial recognition created by Facebook's AI Research (FAIR) team. It uses deep neural networks to precisely detect and validate faces in pictures and videos. Convolutional, pooling, and fully connected layers are among the interconnected neural network layers that make up the DeepFace model [15]. It makes use of a deep architecture that enables it to learn hierarchical representations of faces, taking into account both high-level data like facial landmarks and expressions as well as low-level features like edges and textures [16]. This model takes a bit of an advanced approach than others by adding the transformation and piece-wise affine transformation in the procedure, and the algorithm is empowered for delivering more accurate results [17]. Once trained, the DeepFace model is capable of verifying and recognizing faces. It generates high-dimensional feature vector that represents the face from an input image. To establish whether two faces belong to the same person, this feature vector can then be compared to other

**Table 4** DeepFace model architecture

Layer	Output shape	Number of parameters
Input	(Batch,3,224,224)	0
Conv3-64	(Batch,64,112,112)	1,792
MaxPooling2D	(Batch,64,56,56)	0
Conv3-128	(Batch,128,56,56)	73,856
MaxPooling2D	(Batch,128,28,28)	0
Conv3-256	(Batch,256,28,28)	295,168
Conv3-256	(Batch,256,28,28)	590,080
MaxPooling2D	(Batch,256,14,14)	0
Conv3-512	(Batch,512,14,14)	1,180,160
Conv3-512	(Batch,512,14,14)	2,359,808
MaxPooling2D	(Batch,512,7,7)	0
Conv3-512	(Batch,512,7,7)	2,359,808
Conv3-512	(Batch,512,7,7)	2,359,808
MaxPooling2D	(Batch,512,3,3)	0
Flatten	(Batch,4608)	18,874,368
Dense	(Batch,4096)	16,781,312
Dense	(Batch,4096)	524,416
Dense (Embedding)	(Batch,128)	0

feature vectors. As mentioned in Table 4. The Deep Face model, spanning 18 layers, employs convolution and pooling operations for feature extraction. It culminates in dense layers that facilitate effective facial embedding. Model complexity is evident through substantial parameter counts.

## 5 Implementation and Results

### 5.1 Comparison Between Models

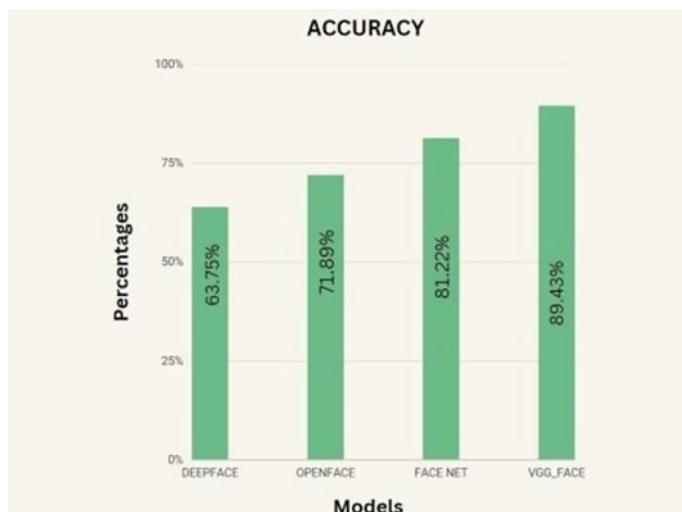
The research paper analyzes the facial recognition models VGGFace, OpenFace, FaceNet, and DeepFace in crime detection. The performance and result of the VGGFace, FaceNet, OpenFace, and DeepFace models trained on the given dataset were evaluated and analyzed. The models were trained using similar configurations, including the same preprocessing techniques and training parameters. The evaluation of the models was carried out on a separate test set, consisting of previously unseen samples. The accuracy metric was used to assess the performance of the models. Table 5 shows the comparison between the models.

**Table 5** Model comparison

Model	Error rate (%)	Accuracy (%)	Precision (%)	Time (s)	Layers	Optimizer
VGGFace	28.35	89.43	68.35	0.3298	19	ADAM
FaceNet	33.67	81.22	63.87	0.4279	9	ADAM
OpenFace	34.92	71.89	54.92	0.3399	14	NADAM
DeepFace	38.11	63.75	58.11	0.3744	18	SGD

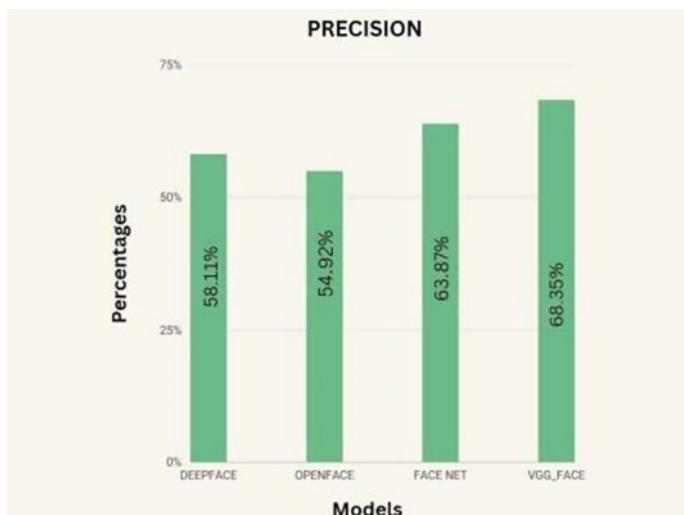
## 5.2 Results

On the presented dataset, the assessment and comparative study of the VGGFace, FaceNet, OpenFace, and DeepFace models showed that VGGFace performs particularly well in facial recognition tasks. It outperformed other models with an excellent accuracy of 89.43% as shown in Fig. 2. It should be noted that VGGFace had the lowest mistake rate, coming in at 28.35% as shown in Fig. 3, demonstrating how well it can reduce misclassification. With an accuracy rate of 68.35%, an important statistic in such challenges, VGGFace demonstrated its aptitude for making precise positive predictions. Its effectiveness was also impressive; one face verification only required 0.32979688 s as shown in Figs. 4 and 5, respectively, making it more suitable for real-time applications. In conclusion, the evaluation highlights VGGFace's exceptional performance across crucial metrics, positioning it as a serious candidate for accurate and effective facial recognition. Numerous practical applications that call for precise and quick identification can greatly benefit from its strong accuracy and efficiency.

**Fig. 2** Accuracy of models



**Fig. 3** Error rate of models



**Fig. 4** Precision of models

### 5.3 Limitations

While the results are promising, there are certain limitations to the comparative comparison of models for face recognition for criminal detection, such as the scarcity of complete and representative datasets for training and evaluation. Because there



**Fig. 5** Time for one face verification

is no commonly accepted standard meter for facial recognition in criminal detection, selecting relevant performance indicators might be difficult. Limited access to specialized gear and processing resources, for example, may limit the volume and complexity of the models and datasets that may be examined. Ethical and legal concerns, including as privacy and data security, must be properly addressed. Model evaluation in controlled research settings may not fully replicate real-world events, limiting the findings' practical usefulness. Furthermore, due to the rapid advancement of facial recognition technology, certain conclusions may become obsolete with the development of new models.

## 6 Key Findings and Contributions

The study paper compares four face recognition models for criminal detection (VGGFace, OpenFace, FaceNet, and DeepFace). The emphasis here is on determining the accuracy and precision of these models. The results show that the VGGFace model outperforms the other three models in terms of accuracy and precision. The VGGFace model outperforms other models in correctly recognizing criminal faces with high accuracy while minimizing false positives. In terms of contributions, the study paper enhances understanding of the comparative performance of facial recognition models in criminal detection. It establishes the efficacy of the VGGFace model and emphasizes the necessity for additional research utilizing larger and more diverse datasets. Furthermore, the paper emphasizes the significance of considering ethical issues as well as real-world applicability. Overall, the research

report provides useful insights into the comparative examination of facial recognition models for criminal detection, with the VGGFace model appearing as the most accurate and precise among the models evaluated.

## 6.1 Future Scope

There are various potential topics for future research on the topic of a comparative examination of facial recognition models for criminal detection using VGGFace, OpenFace, FaceNet, and DeepFace models. Among the future scopes are:

- Integration of several models:** Look at the possible benefits of combining the strengths of numerous facial recognition models to increase overall performance and accuracy in criminal detection scenarios.
- Bias reduction strategies:** Investigate and develop techniques to address and reduce biases in facial recognition models, particularly those related to race, gender, and age, in order to provide fair and unbiased criminal identification.
- Real-world deployment and evaluation:** Conduct research that focus on deploying the facial recognition models in real-world criminal detection settings, taking into account environmental characteristics, lighting conditions, and scalability. Evaluate their performance and usefulness in real-world circumstances.
- Robustness against adversarial assaults:** Investigate facial recognition models' vulnerability to adversarial attacks, such as spoofing or manipulation efforts, and propose robust approaches to improve their resistance and reliability.

## 7 Conclusion

The research paper analyzes the facial recognition models VGGFace, OpenFace, FaceNet, and DeepFace in crime detection. The study assesses their performance and discovers that the VGGFace model has the best accuracy of 89.43%. The accuracy of FaceNet, OpenFace, and DeepFace model is 81.22%, 71.89%, and 63.75%, respectively. The findings help us understand how effective these models are in criminal identification. The study demonstrates the VGGFace model's potential for improving face recognition technology in criminal detection applications. Overall, the research paper provides useful insights into the comparative examination of facial recognition models for criminal detection.

## References

1. Aherwadi, Nagnath B., Deep Chokshi, Dr Sagar Pande, and Aditya Khamparia (2021) Criminal identification system using facial recognition. In: Proceedings of the international conference on innovative computing communication (ICICC)

2. Chhoriya, Piyush (2019) Automated criminal identification system using face detection and recognition. *Int Res J Eng Technol (IRJET)* 6(10)
3. Serra, Xavier, and Javier Castán (2017) Face recognition using deep learning. Polytechnic University of Catalonia, Catalonia 78
4. Fuad MTH, Fime AA, Sikder D, Iftee MAR, Rabbi J, Al-Rakhami MS, Islam MN (2021) Recent advances in deep learning techniques for face recognition. *IEEE Access* 9:99112–99142
5. Ban, Jozef et al (2013) An automatic training process using clustering algorithms for face recognition system. In: *Proceedings ELMAR-2013*. IEEE
6. Poses: U. Scherhag, R. Raghavendra, K.B. Raja, M. Gomez-Barrero, C. Rathgeb, C. Busch (2017) On the vulnerability of face recognition systems towards morphed face attacks, in 2017 5th International Workshop on Biometrics and Forensics (IWBF), pp 1–6VGGface
7. He, Mingjie, et al (2020) Deformable face net for pose invariant face recognition. *Pattern Recognition* 100:107113
8. Amos B, Ludwiczuk B, Satyanarayanan M (2016) Openface: a general-purpose face recognition library with mobile applications. *CMU School of Computer Science* 6(2):20
9. Wang M, Deng W (2021) Deep face recognition: A survey. *Neurocomputing* 429:215–244
10. Schroff, Florian, Dmitry Kalenichenko, and James Philbin (2015) FaceNet: a unified embedding for face recognition and clustering. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE
11. Guo, Tianmei, et al (2017) Simple convolutional neural network on image classification. In: *2017 IEEE 2nd International conference on Big Data analysis (ICBDA)*. IEEE
12. Santoso, Kevin, and Gede Putra Kusuma (2018) Face recognition using modified Open-Face. *Procedia Comput Sci* 135:510–517
13. Song L et al (2019) Occlusion robust face recognition based on mask learning with the pairwise differential siamese network. In: *Proceedings of the IEEE/CVF international conference on computer vision*, 2019
14. N. Bonettini , Cannas, Edoardo Mandelli, Sara Bondi, Luca Bestagini, Paolo Tubaro, Stefano et al. (2021) Video face manipulation detection through ensemble of CNNs. In: *International conference on pattern recognition (ICPR)* Milan, Italy, Jan 10–15, 2021, pp 5012–5019
15. Elmahmudi A, Ugail H (2019) Deep face recognition using imperfect facial data. *Futur Gener Comput Syst* 99:213–225
16. Du Hang et al (2022) The elements of end-to-end deep face recognition: a survey of recent advances. *ACM Comput Surv (CSUR)* 54.10s:1–42
17. Parkhi OM, Vedaldi A, Zisserman A (2015) Deep face recognition. In: *British machine vision conference*, p 6

# Transcription of Ancient Indian Manuscripts Through Artificial Intelligence—Current Status of Technology and the Way Forward



R. Harish and G. N. Raghavendra Rao

**Abstract** India holds the largest known collection of ancient and medieval manuscripts, but many of these still remain unread and uncatalogued due to the immense amount of time and effort required to transcribe them manually. Automating this process would be highly beneficial, as it would reveal valuable information about our past that is currently hidden within these manuscripts. While these documents are written in various languages and scripts, Sanskrit is the most common. This paper takes a practical approach to explore the feasibility of using machine learning to transcribe these manuscript treasures using publicly available information on current technology. Some work has already been done on European platforms such as Transkribus and eScriptorium to transcribe texts in Sanskrit, Pali/Prakrit, and Tamil. It is possible to build on these efforts to develop a viable system for transcribing Indian manuscripts on a large scale, but this would require a dedicated effort to build and train models that can handle the complexity of these manuscripts with adequate accuracy and speed. While a deep learning model built by a private commercial enterprise in the USA called Nanonets has been found to be the most user-friendly and can provide transcriptions of thirteenth/fourteenth century Sanskrit and Tamil manuscripts without any training or visual enhancement of images, it is not yet adequately accurate and may be more expensive. Regardless of the method chosen, this project will require financial and technical support from government and/or private entities. The Government of India's National Mission for Manuscripts could be the focal point for coordinating this activity, and the effort would likely be well worth it.

**Keywords** Manuscript · Transcription · Machine Learning · Transkribus · eScriptorium · Nanonets

---

R. Harish (✉) · G. N. Raghavendra Rao  
Faculty Members at ICFAI Business School, Bangalore, India  
e-mail: [harish@ibsindia.org](mailto:harish@ibsindia.org)

An Off-Campus Center of the ICFAI Foundation for Higher Education (IFHE), Hyderabad, India

## 1 Motivation for the Study

India is a diverse country with many traditions, languages, and a rich cultural history. As a result, it has a vast repository of various religious and secular literature, historical and administrative records in multiple languages and scripts. These are in the form of handwritten manuscripts, which run into millions. India most likely has the largest number of manuscripts in the world. The manuscripts are in various sizes and shapes and are on a variety of substrates such as palm leaves, birch bark, handmade paper, cloth, and leather. The manuscripts are stored in numerous archives in different parts of the country.

The Government of India's Ministry of Tourism & Culture has a National Mission for Manuscripts initiated in 2003. Its website states that 70% of the manuscripts are in Sanskrit, and the rest in other languages, including Prakrit, Pali, Tamil, Persian, etc. These are written in various scripts—Brahmi, Sharada, Modi, Grantham, Tigalari, Naskh, Nasta'liq, and many others. The manuscripts fall largely into two categories—(1) historical and administrative documents and (2) those pertaining to religion, literature, and culture. The Mission has also digitized manuscripts, and more than 300,000 manuscripts running into over 33 million pages have been digitized so far.

However, many ancient texts still remain unread due to the time and effort required to transcribe and read them. Valuable information is waiting to be discovered, but there is shortage of scholars to document, catalogue, edit, translate, and publish them. Can machine learning be used to automate this process? This study explores the current status of technology in this area, particularly in the context of Indian scripts, and what further needs to be done.

## 2 Handwriting Recognition Systems—A Brief Overview

Two main types of handwriting recognition (HWR) systems exist: online and offline. Online tracks pen movements, while offline scans and converts handwritten text. We focus on offline transcription of manuscripts. Handwritten character recognition is difficult due to diverse writing styles and cursive text, which can be hard to decipher. Recent advancements in HWR systems have been achieved through the use of artificial neural networks and deep learning techniques. These methods have greatly improved the accuracy of recognizing handwritten characters, leading to a notable overall improvement in HWR systems.

HWR technology is improving and becoming more widely used in various applications. Deep learning, which uses artificial neural networks, can enhance the accuracy and performance of HWR systems by allowing models to learn from examples and experiences. Two types of deep learning models are mainly used for handwriting recognition: convolutional neural networks (CNNs) and recurrent neural networks (RNNs). CNNs are well-suited for tasks that involve processing spatial data, such

as images. RNNs are better suited for tasks that involve processing sequential data, such as text.

### **3 Problems and Issues Related to Handwritten Character Recognition**

This section presents an overview of the general problems and issues encountered in the context of handwritten character recognition (HCR) and machine transcription of old manuscripts.

Transcribing historical manuscripts is the most difficult of the HCR domains. Many national archives and other organizations worldwide are pursuing efforts toward handwriting recognition of historical documents [1], as this will make the texts more widely available and usable. However, automated character recognition of old manuscripts can only provide a partial solution, but can make the process easier and faster, because old manuscripts are far more complex and varied than modern handwriting.

Recognizing handwriting in old documents is difficult due to variations in character styles and noise elements. Most optical character recognition (OCR) systems require context to recognize text. Reading handwritten text requires specialized knowledge of the subject domain, like reading a doctor's prescription requires knowledge of medicines. Further, automatic transcriptions often lead to considerable distortion from the original. For example, different letter forms used in the past are normalized into one when transcribing medieval European texts. Though Medieval Unicode Font Initiative (MUFI) characters are in the public domain, they require additional work for proper rendering and can pose problems when transcribing plain text.

### **4 Literature Survey—Developments in Machine Transcription of Old Manuscripts in European Scripts**

To understand machine reading of Indian scripts, it is useful to look at how it is being used in other languages. European languages have made significant advancements in this technology. Experts have written papers, articles, and blogs on this topic, which we will briefly highlight.

#### **4.1 Deciphering Unknown Scripts**

Deciphering an unknown script is a challenge that requires identifying the underlying language. This was crucial in deciphering hieroglyphics (Coptic), Linear B (Greek),

and Mayan glyphs (Ch’olti’). However, scripts like Linear A and Indus Valley remain undeciphered as the languages are not known [2]. Deciphering them through machine learning is nearly impossible without identifying the language(s).

## 4.2 *Decoding the Voynich Script*

The Voynich manuscript is an old book from the fifteenth century that has around 240 pages. The book was written in a language that was evidently known, but it was undecipherable as it used substitution ciphers and anagrams. Researchers from the University of Alberta in Canada used artificial intelligence to try to understand the code [2]. They used a method called algorithmic decipherment. After testing the algorithm on 380 different translations from a set of candidate languages, they concluded that the language of the coded text was most likely Hebrew. However, the transcribed sentences did not make much sense when the words were put together. So, more research is needed to understand the book.

## 4.3 *Start-Flow-Read (SFR) Model*

The Start-Follow-Read (SFR) model was introduced to achieve better accuracy in handwriting recognition of historical documents [1]. The authors’ CNN-LSTM network achieved improved performance by addressing line segmentation errors, which were often more common than character recognition errors. Three sub-elements included a Start-of-Line (SOL) finder, a Line Follower (LF), and a line-level HWR model. The model worked even better with a language model or lexicon to refine predictions.

## 4.4 *‘In Codice Ratio’ Project to Transcribe Vatican Secret Archives*

‘In Codice Ratio’ [3] was a project that used NVIDIA Quadro GPU and convolutional neural networks to transcribe Vatican Secret Archives containing medieval Latin manuscripts. The system achieved 96% accuracy in recognizing handwritten characters with minimal training effort and produced good transcriptions for scalability. The approach required minimal training effort, and the training data was produced by 120 high school students. The system produced good transcriptions, which could be used to speed up the transcription and make it more scalable.

#### **4.5 ‘Normalized’, ‘Semi-diplomatic’, and ‘Diplomatic’ Manuscripts—Guidelines for Transcribing the Paris Bibles**

Transcribing medieval Latin manuscripts, particularly Paris Bibles from the thirteenth and fourteenth centuries, posed unique challenges [4]. The authors identified three types of transcription and noted that different models can lead to significant bias. Fully automated ‘normalized’ transcriptions can distort the original, while ‘diplomatic’ transcriptions can be too complex for non-scholarly readers. The authors suggested defining different guidelines before transcription depending on the end use, with appropriate character sets for training the models. They also noted that popular HCR systems do not address these issues.

#### **4.6 Transcribing Documents from the Abbey Library of St. Gall**

A University of Notre Dame team developed a machine learning model to transcribe and record handwritten documents from the Abbey Library of St. Gall in Switzerland [5]. The manuscripts are handwritten in medieval Latin on parchment paper and date back to the eighth century. The project aimed to create a quick, searchable reading of the text using visual psychophysics to mimic the perception of a page through the eyes of a human reader. The team used the cuDNN-accelerated PyTorch deep learning framework and GPUs to train the neural networks with experts’ transcriptions. The program was also adapted to transcribe Ethiopian texts. However, many improvements are required, including dealing with damaged and incomplete documents, illustrations, and abbreviations.

### **5 Literature Survey—Academic Research Papers on Handwritten Character Recognition of Indian Scripts**

The literature survey discusses the progress made in handwritten character recognition (HCR) systems for Indian scripts. European languages have simpler scripts, making HCR easier, while Indian scripts have hundreds of distinct symbols with vowel modifiers and mixed consonants. However, the advancement in HCR is quite significant even in the case of Chinese, Japanese, and Korean, though their scripts are very complicated. The work on Indian scripts lags considerably behind [6].

As mentioned earlier, European scripts are far simpler compared to Indian scripts. For instance, the English (Latin) script has only 52 characters to represent letters (26 small and 26 capital letters), 10 numerals, some punctuation marks, and special characters. In contrast, because of vowel modifiers and mixed consonants, Indian

scripts have hundreds of distinct symbols. But there are ways to code these in a relatively simple way.

The Devanagari script has three components for each letter, making it difficult to segment characters. Some letters and numerals are very similar, making it easy to mistake them for one another when written by hand [6]. This complexity makes machine reading of Indian scripts more difficult than European scripts. And OCR systems for Hindi and other Indian languages have not demonstrated high accuracy due to the complexity of the characters.

Similar-looking characters in the Devanagari script cause classification errors. To address this problem, the use of Fisher linear discriminant model was proposed to detect the critical region and extract additional features. This technique improved accuracy when tested on a database of 36,172 handwritten Devanagari characters [7].

A detailed review of the developments in handwritten character recognition including several Indian languages indicated that Devanagari characters have not been explored as much as English. Progress has been made even in complex languages, such as Japanese, Chinese, and Korean, but accuracy still lags behind English. They used a dataset of 92,000 images to train a deep learning model, achieving 98.13% accuracy [8].

A review of 42 papers on recognizing characters in Devanagari/Hindi script shows that CNN and SVM are popular deep learning methods with high accuracy results [6]. However, most studies only use simple handwritten characters, and accuracy varies based on the dataset. The research suggests further exploration with more extensive and complex datasets.

In another study, transfer learning was used to improve identification of Devanagari characters. They tested pre-trained CNN models and found that Inception V3 provided the highest accuracy of 99%, while AlexNet was the fastest. The dataset consisted of 92,000 images [9]. Yet another study achieved a 99.40% recognition accuracy in Bangla character recognition using various CNN techniques on a dataset of 10,000 samples [10].

Most research papers on transcribing Indian scripts only focus on basic characters and report high accuracy levels. However, practical solutions for transcribing handwritten documents and ancient manuscripts are still lacking. European and US projects have made far better progress and can serve as starting points for further developments.

## **6 AI-Powered Platforms for Text Recognition, Transcription, and Historical Document Search—An Overview**

AI-powered platforms transcribe old manuscripts, recognize text, and search it for keywords. These tools are useful for historians and researchers to access information in historical documents.

### ***6.1 Workings of an AI-Powered HWR Platform***

OCR and ML are combined in AI-powered platforms for text or handwriting recognition, transcription, and document searching. OCR converts images to machine-readable text, while ML allows models to learn from data, improving OCR's accuracy. The transcribed text can be searched for specific information, key words, or phrases [11].

### ***6.2 Benefits of an AI-Powered HWR Platform***

AI-powered platforms for text recognition, transcription, and historical document search have several benefits, including higher accuracy levels compared to traditional OCR methods, faster transcription, scalability for large volumes of historical documents, and increased accessibility for researchers and the public. These benefits can improve historical literacy and understanding while saving time and money.

### ***6.3 Challenges of Using an AI-Powered HWR Platform***

Using AI-powered platforms for text recognition, transcription, and searching of historical documents can present challenges such as high development and maintenance costs, difficulty in obtaining the large amounts of data required to train ML models and potential inaccuracies. Despite these challenges, AI-powered platforms can be valuable tools for accessing and transcribing historical documents, but require significant effort in training the ML models.

## 7 Advantages of Deep Learning Techniques Over Regular Neural Networks

For OCR technology, deep learning techniques are superior to traditional neural networks because they can learn more complex patterns and be trained on larger datasets. Deep learning techniques are used in various applications, including document scanning, optical character recognition, and handwriting recognition. Examples of deep learning techniques used for OCR include convolutional neural networks, recurrent neural networks, and transformers. As deep learning technology continues to develop, we can expect to see even more accurate and efficient OCR systems being developed.

## 8 Prominent Handwriting Text Recognition Systems in Actual Use

Some handwriting text recognition systems are tailored for transcribing old manuscripts and can guide developing models to transcribe Indian manuscripts. Basic versions may be free, but advanced versions may require registration, commitment, participation, or payment. The HTR workflow includes importing a text image, segmenting it into words and lines, transcribing it automatically with manual corrections, and exporting the transcribed text or model. The article examines four popular handwriting recognition systems in practical use.

## 9 OCropus API

OCropus is a free document analysis and OCR system that is language agnostic and can handle various languages and scripts [12]. OCropus API allows fast and accurate OCR solutions for any use case, with over 100 languages and scripts to choose from. However, OCropus requires image preprocessing and appropriate model selection and does not provide a GUI or web service. OCR.space is a possible alternative but has a limit of 500 daily requests per IP address. Therefore, OCropus would not be ideal for transcribing old Indian manuscripts on a large scale.

## 10 eScriptorium

eScriptorium is a platform for transcribing historical scripts and languages. It offers automatic and manual transcription, model training, and import/export formats [13]. It is built on Kraken, an OCR-based HTR engine optimized for historical and non-Latin scripts. It has received funding from the RESILIENCE project and The Andrew W. Mellon Foundation and encompasses several projects. It has partner projects on Islamic texts, Hebrew texts, Vietnamese inscriptions, and French Notarial Records. It has an extensive blog and tutorial site but requires access through partner projects. Other teams contribute to eScriptorium through add-ons to improve accuracy. It can be used to transcribe Indian manuscripts through a partner project.

## 11 Transkribus

Transkribus is an AI-powered platform for digitization, text recognition, transcription, and historical document search. It is a free and open-source platform that anyone can use regardless of technical expertise [14]. It was funded through the READ project under the EU's Horizon 2020 initiative. The platform can generate data from handwritten and printed texts and is based on artificial neural networks. It can handle several European languages and scripts and be trained for other languages [15]. The platform has more than 100,000 members and has processed over 40 million pages. Transkribus uses various AI techniques such as OCR, machine learning, and NLP to improve text recognition accuracy. Users can train their own AI models or use public models that are available for different languages and scripts. The platform suits researchers, archivists, librarians, genealogists, and citizen scientists working with historical documents.

### 11.1 Benefits of Using Transkribus

Transkribus uses AI to improve text recognition accuracy, saving time and money while preserving historical document integrity. It also simplifies collaboration and data sharing, speeding up transcription and analysis, and increases accessibility, promoting research and education.

## ***11.2 Limitations of Transkribus***

Transkribus can struggle with difficult handwriting or damaged documents, is a paid service, is complex for users unfamiliar with AI, and struggles with recognizing different varieties of English handwriting.

## ***11.3 Some Tips for Using Transkribus Effectively***

To improve accuracy, high-quality scans should be used, and errors should be corrected to ensure accuracy. Annotation tools can also be used to make documents more searchable and understandable. Working with others on the platform can also help improve accuracy and speed up the transcription process.

# **12 Nanonets**

Nanonets is an AI-based platform that automates document workflows and financial controls using machine learning APIs. It raised \$10 million in February 2022 to expand its engineering and AI/ML teams and upscale its operations and marketing teams in new geographies [16]. Nanonets uses OCR, machine learning, and deep learning algorithms to extract relevant information from unstructured and semi-structured documents. Nanonets is primarily designed for businesses and developers who need to automate data capture and processing from many documents. Its AI engine can comprehend documents, update databases, validate information, and execute business decisions without human intervention and with high accuracy [17]. Nanonets' customers include governments and enterprises worldwide, including Fortune 500 companies.

## ***12.1 Advantages of Nanonets***

Nanonets offers high accuracy rates for text recognition, transcription, and document searching, even for poor-quality images and PDFs. It can quickly extract text from large collections of documents, saving time and effort for researchers and historians. Additionally, it can process large collections of documents, allowing access and analysis of information that would not be possible using traditional methods.

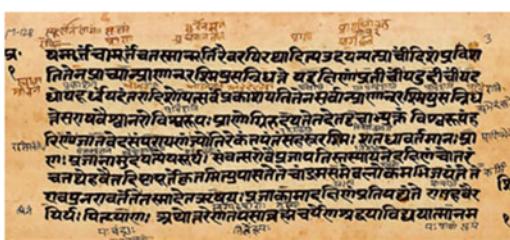
## 12.2 Limitations of Nanonets

Nanonets is a cloud-based platform that offers text recognition, transcription, and searching of historical documents [18]. While its accuracy is imperfect, it can transcribe contemporary handwriting with around 90% accuracy in Indian languages, including Sanskrit, Hindi, Kannada, and Tamil. Nanonets can be expensive, and the cost depends on the number of documents to be processed, the type of text to be recognized, and the level of accuracy required. However, it offers various pricing plans to fit different budgets and has a team of experts available to help users troubleshoot any problems. Overall, Nanonets is a powerful and versatile tool for transcribing various historical documents in diverse languages and scripts.

## 13 Comparison Between Transkribus and Nanonets

We tried converting the same documents in Transkribus and Nanonets for comparison purposes. Below are the observations. Nanonets, an OCR platform that uses deep learning technology, is more accurate than traditional OCR systems like Transkribus. Nanonets' deep learning algorithm can learn the nuances of human handwriting, resulting in higher accuracy. Nanonets provides a web-based interface accessible from any browser, while Transkribus offers both a web-based interface and a desktop application with different features and functionalities. Overall, Nanonets and Transkribus have different strengths and weaknesses, depending on the type of documents and use cases they target. The results on the expert version of Transkribus (which was not accessible to us) could have been better. Figures 1, 2, 3, and 4 in Sect. 15 below provide examples of transcriptions done on Nanonets and Transkribus.

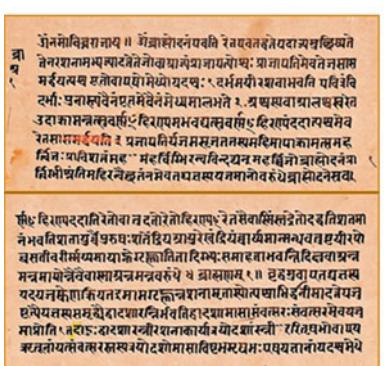
In summary, Nanonets and Transkribus are two OCR platforms with different strengths and weaknesses, depending on the type of documents and the use cases they target.



यमायामवत्सानसातिरेवदग्निरभादिलऽदयन्त्याचीदिशप्रविश  
तिरेनश्चायाप्राणनरथिष्पुरानिधनेयहातीयोदीयोपदमो  
धोयदत्तादेशोपसर्वकायायतिरेनसावीकायत्रिप्रियसनिध  
नेसारावेषानलोविमासृःयाप्राणिरुपतेतदायाभुक्तिविश्वलः  
रिष्टातिवेदसपराणाज्ञातिरेकेषपरहरयःयात्मावत्तीमानः  
एःपूजानामृत्येयसीयसनसरोवेष्प्राणापतिस्तमायनेदद्युष्मोत्ते  
चन्द्रेहेतुदिष्टापतितमित्युपासतेत्याग्नमसमेवलोकमभिजयेत्येम  
रावपुनरात्मीतसादेताक्षण्याप्राजाकामाद्युष्मप्रतिगहोराजाहर  
विभूषित्यापाणधानगतसाप्रवर्येरगाकाविद्यायामानम्

नितेन लक्षणम्

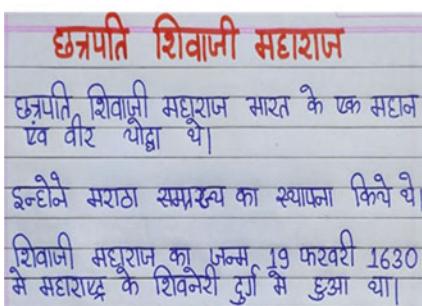
**Fig. 1** Transcription of Pre-14th Century Prashna Upanishad sample manuscript using Nanonets



**Fig. 2** Transcription of 13th Century Shatapatha Brahmana, 14th Khanda, Prapathaka 3 and 4, Sanskrit, Devanagari script using Nanonets



**Fig. 3** Transcription of a Tamil palm leaf manuscript using Nanonets



Transcription by Nanonets

ஷந்தி ஶிவாஜி மஹாராஜ  
ஷந்தி ஶிவாஜி மஹாராஜ மூலம் பதினாறாம் காலத்தில் சிவாஜி மஹாராஜ என்று அழைக்கப்பட்டிருந்தார். இவர்கள் முழுமொழியில் உயிர்போட்டு வாழ்ந்தன.

Transcription by Transkribus

ஷந்தி ஶிவாஜி மஹாராஜ  
ஷந்தி ஶிவாஜி மஹாராஜ மூலம் பதினாறாம் காலத்தில் சிவாஜி மஹாராஜ என்று அழைக்கப்பட்டு வாழ்ந்தார். இவர்கள் முழுமொழியில் உயிர்போட்டு வாழ்ந்தன.

**Fig. 4** Comparison of transcription of contemporary Indian handwriting by Nanonets and Transkribus

## 14 Conclusions

The paper explores the practicality of machine-transcribing ancient Indian manuscripts. Four platforms were considered, with OCROpus being the least convenient. Two AI-driven platforms, eScriptorium and Transkribus, were both developed for transcribing old manuscripts. However, Nanonets was found to be more accurate and user-friendly for transcribing documents in large numbers. Transkribus must be trained for new scripts, but can be easily used for larger sets of documents. The final choice should be based on volume, cost, and available time and manpower. Financial and technical support from government or private entities is required, and the National Mission for Manuscripts could coordinate the activity. It could be well worth the effort.

## 15 Exhibits

### 15.1 *Figure 1*

### 15.2 *Figure 2*

### 15.3 *Figure 3*

### 15.4 *Figure 4*

## References

1. Wigington C, Tensmeyer C, Davis B, Barrett W, Price B, Cohen S (2018) Start, follow, read: end-to-end full page handwriting recognition. In: Computer Vision Foundation. eccv 2018 Springer
2. Hauer B, Kondrak G (2016) Decoding anagrammed texts written in an unknown language and script. *Trans Assoc Comput Linguist* 4:75–86
3. Firmani D, Merialdo P, Nieddu E, Scardapane S (2017) In Codice ratio: OCR of handwritten Latin documents using deep convolutional networks. In: Proceedings of the 11th international workshop on artificial intelligence for cultural heritage (AI\*CH 2017)
4. Guéville E, Wrisley DJ (2022) Transcribing medieval manuscripts for machine learning. Cornell University—Computer Science—Digital Libraries. <https://arxiv.org/abs/2207.07726>, [https://www.researchgate.net/publication/362089873\\_Transcribing\\_Medieval\\_Manuscripts\\_for\\_Machine\\_Learning/link/62d625f5bf4b98532233d9f3/download](https://www.researchgate.net/publication/362089873_Transcribing_Medieval_Manuscripts_for_Machine_Learning/link/62d625f5bf4b98532233d9f3/download)
5. Horton M (2021) Deciphering ancient texts with AI. Nvidia Developer, Technical Blog. <https://developer.nvidia.com/blog/deciphering-ancient-texts-with-ai/>.
6. Agrawal M, Chauhan B, Agrawal T (2022) Machine learning algorithms for handwritten Devanagari character recognition: a systematic review. *J Sci Technol* 7(1)
7. Jangid M, Srivastava S (2018) Handwritten Devanagari similar character recognition by fisher linear discriminant and pairwise classification. *Int J Image Graph* 18(04)

8. Bhardwaj A, Singh R (2020) Handwritten Devanagari character recognition using deep learning—Convolutional neural network (CNN) model. *PalArch's J Archaeol Egypt/Egyptol* 17(6)
9. Aneja N, Aneja S (2019) Transfer learning using CNN for handwritten Devanagari character recognition. In: IEEE international conference on advances in information technology (ICAIT)
10. Sen S, Shao D, Paul S, Sarkar R, Roy K (2018) Online handwritten Bangla character recognition using CNN: a deep learning approach. *Intell Eng Inf Adv Intell Syst Comput (AISC)* 695:413–420
11. Sphærber G (2018) A gentle introduction to OCR. Towards Data Science. <https://towardsdatascience.com/a-gentle-introduction-to-ocr-ee1469a201aa>.
12. Bruel T (2007) Announcing the OCropus open source OCR system. Google for Developers. <https://developers.googleblog.com/2007/04/announcing-ocropus-open-source-ocr.html>
13. eScriptorium: a digital text production pipeline for print and handwritten texts using machine learning techniques. <https://escriptorium.openiti.org/>
14. Transkribus—Unlock historical documents with AI. Read Co-op. <https://readcoop.eu/transkribus/>
15. Kahle P, Colutto S, Hackl G, Muhlberger G (2017) Transkribus—A service platform for transcription, recognition and retrieval of historical documents. In: 4th IAPR international conference on document analysis and recognition (ICDAR), Kyoto, Japan, pp 19–24, <https://doi.org/10.1109/ICDAR.2017.307>. <https://ieeexplore.ieee.org/document/8270253>. Griffiths R (2022) Transkribus in practice: abbreviations. The Digital Orientalist. <https://digitalorientalist.com/2022/11/01/transkribus-in-practice-abbreviations/>
16. PR Newswire (2022) Nanonets raises \$10M from elevation capital to help global enterprises automate their document workflows using AI. CISION—PR Newswire. <https://www.prnewswire.com/news-releases/nanonets-raises-10m-from-elevation-capital-to-help-global-enterprises-automate-their-document-workflows-using-ai-301483676.html>. Accessed 16 Feb 2022
17. Agarwal R (2022) Nanonets. Deep learning-based OCR for text in the wild. <https://nanonets.com/blog/deep-learning-ocr/>
18. Nanonets' user platform. <https://app.nanonets.com/#/models>

# Computer Vision and Convolutional Neural Network for Dense Crowd Count Detection



D. Sirisha , S. Sambhu Prasad , and Subodh Kumar

**Abstract** Crowd counting is an approach for the process of counting the people in an image. Extensive studies on crowd detection and density estimation are being carried out for crime prevention, crowd irregularities, public safety, visual monitoring, and urban planning. Approaches to detect crowd count are available in the literature; however, available algorithms could not detect the accurate number of people. Therefore, in the current work, computer vision techniques in fusion with convolutional neural networks (CNNs) are employed to produce impressively precise estimates. The proposed work will precisely detect count of the persons in an image using computer vision and CNN. Pattern recognition techniques are employed for crowd count detection by using face detection. However, detecting a face in the crowd is complex as inconsistency prevails in human faces comprising of color, pose, expression, position, orientation, and illumination. Congested Scene Recognition Network (CSRNet) attains 47.3% lower mean absolute error compared with existing techniques. The current work is also extended to various intended applications such as vehicles. The experimental results reveal that CSRNet has shown significant improvement in the output by 15.4% better MAE than existing contemporary approaches.

**Keywords** Deep convolutional neural networks · Crowd counting · Density detection · Faster R-CNN

---

D. Sirisha

Department of Computer Science Engineering, Nadimpalli Satyanarayana Raju Institute of Technology, Visakhapatnam, A.P., India  
e-mail: [sirishad998@gmail.com](mailto:sirishad998@gmail.com)

S. Sambhu Prasad

Department of Mechanical Engineering, Nadimpalli Satyanarayana Raju Institute of Technology, Visakhapatnam, A.P., India

S. Kumar

Department of Electronics and Communication Engineering, Pragati Engineering College, Surampalem, A.P, India

## 1 Introduction

Estimating the population in a given area is referred to as crowd counting. The increase in the population is leading to rapid urbanization and crowd gatherings. Hence, CV-based crowd analytics is profoundly applied. Additionally, algorithms which aimed at crowd analytics are also effective in fields such as agriculture monitoring and environmental survey. With the advent of novel networks, crowd counting techniques showed improvement.

To avoid the spread of the viral diseases, it is essential to maintain social distancing. It will help maintaining distance between each individual to cut down rate of the spread of the diseases. Crowd count is especially beneficial to gage the crowded places such as function halls and schools. Social distancing benefits the people suffering with pre-medical conditions. This model can identify heavy crowded places. This model can also be used in traffic analysis and vehicle counting which helps to avoid traffic jams. Crowd counting is crucial for numerous real applications, such as traffic control, security, and disaster management.

Several approaches to count the crowd progressed from fully connected layers to the complex architectures. In this paper, CSRNet architecture is used for crowd computing.

In the present work, a novel approach for crowd counting is presented using computer vision and CNN. The current article is further classified as follows. In Sect. 2, existing work available for counting crowd is detailed. In Sect. 3, proposed model for counting crowd is introduced. Section 4 presents performance analysis of the proposed approach with widely referred existing approaches, and Sect. 5 presents conclusions of the present work with an outline on the future scope.

## 2 Literature Survey

Several approaches are employed to count the persons in a particular image. These approaches are detailed below.

### 2.1 *Detection-Based Approaches*

Past research relied on detection-based methods that involved a moving window-like detector for counting the individuals. These techniques demand trained classifiers for extracting features from entire human body, Switch Haar wavelets and histogram-oriented gradients. These provide low performance when the images consist of more number. Recent research motivated on improving the method by focusing on detecting specific body parts for counting the people in the crowd.

## 2.2 Regression-Based Approaches

As detection-based methods are unacceptable when the image has high count of people, researchers made an attempt to apply regression techniques for learning the relationships between extracted features from cropped images for calculating the number of people.

## 2.3 Density Estimation-Based Approaches

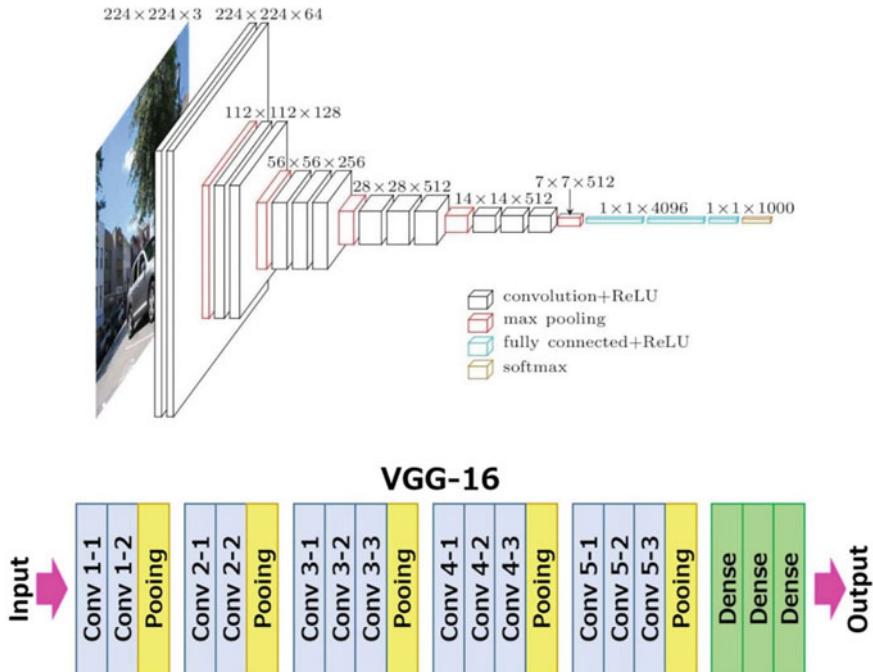
Researchers identified that regression methods ignored a crucial feature, named saliency which resulted in errors in local regions. To circumvent this problem, linear mapping can be done between the local area's characteristics and its object density mappings. Such that saliency is integrated during learning process. As ideal linear mapping is hard to attain, random forest regression is employed to learn nonlinear mapping instead of linear one.

## 2.4 Crowd Detection Using CNN

Boominathan et al. [1], Marsden et al. [2], Sindagi et al. [3], and Bhangale et al. [4] proposed works demonstrating that CNN-based approaches outperform all the approaches mentioned above in Sects. 2.1–2.3. A simple CNN-based crowd counting model counts the individuals in a given image. Some of the prominent CNN models used in crowd counting are multi-column-based architecture (MCNN) proposed by Zhang et al. [5], and Sam et al. [6] proposed Switching-CNN (SCNN).

## 3 Proposed Model

The proposed design's main idea is to use a deeper CNN to capture high-level features having bigger receptive fields to generate high-quality density maps without increasing network convolution which is also studied by Shen et al. [7]. This network architecture is specifically designed to examine highly congested scenes. The architecture of VGG-16 is shown in Fig. 1.



**Fig. 1** Architecture of VGG-16

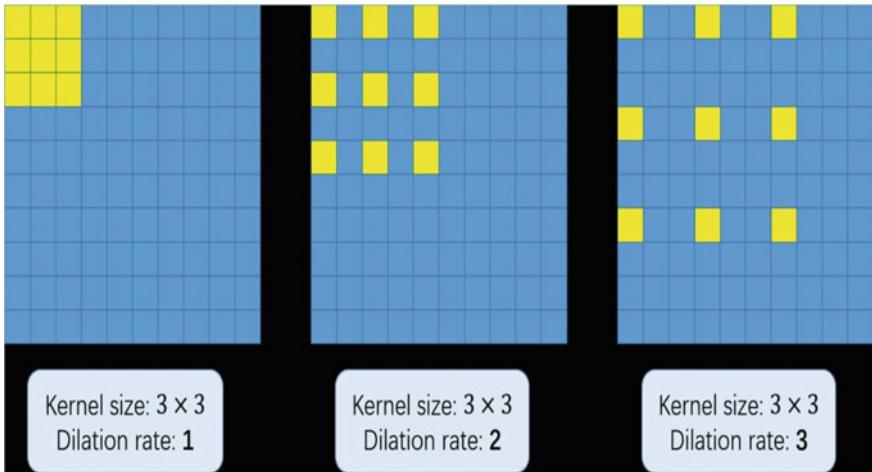
### 3.1 Congested Scene Recognition Network (*CSRNet*)

CSRNet is an algorithm/model that is designed specifically to examine the congested scenes. This architecture can be effectively applied to the congested scenes. CSRNet uses VGG-16 network as its front-end and dilated CNN as its back-end.

### 3.2 VGG-16

Simonyan et al. [8] proposed CNN VGG-16, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, and achieved 92.7% accuracy in ImageNet, for a dataset involving above 14 million images with 1000 classes. By consecutively replacing numerous 33 kernel size filters for significant number of large kernel size filters, this model outperformed AlexNet. Using NVIDIA Titan Black GPUs, VGG-16 underwent weeks of training. The architecture of VGG-16 is presented in Fig. 1.

In the current work, VGG-16 is chosen as front-end of CSRNet due to its robust capability to learn and malleable architecture for focusing on the back-end to generate crowd density.



**Fig. 2**  $3 \times 3$  convolution kernels with different dilation rate as 1, 2, and 3

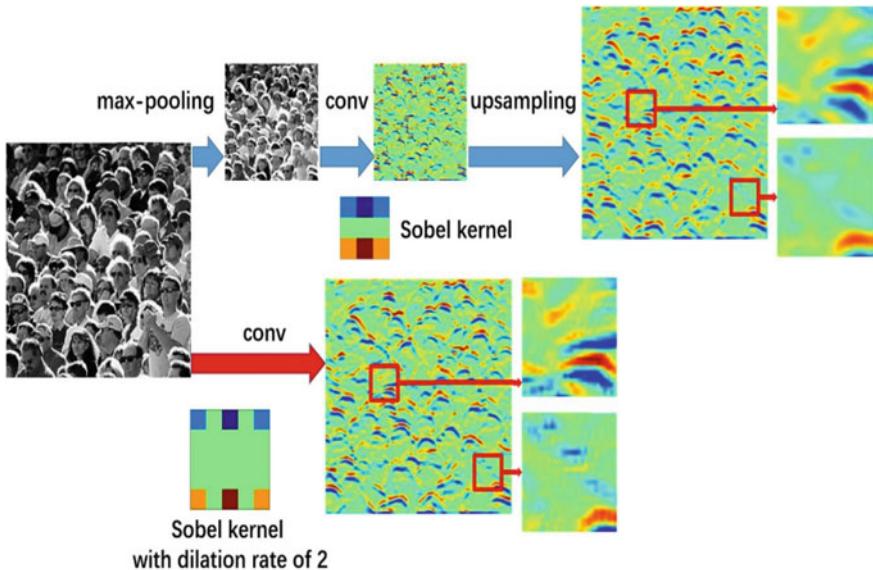
### 3.3 *Dilated Convolution*

Yuhong et al. [9] studied that dilated convolutional layers have proven to attain considerable progress in accuracy for segmentation tasks and observed to be notable alternate of pooling layer. Maximum and average pooling layers are frequently employed to sustain invariance and control overfitting, although lessen spatial detail of feature maps. De-convolutional layers are able to reduce information damage; however, they may not be appropriate in all situations due to their increased complexity and execution latency. Figure 2 shows  $3 \times 3$  convolution kernels with different dilation rate as 1, 2, and 3.

A superior option is dilated convolution, which alternates the pooling and convolutional layers using sparse kernels. The receptive field is expanded by this character without the complexity or processing load increasing. Figure 3 presents the comparison of maxpooling, convolution, and upsampling.

### 3.4 *Setting Up Network*

In the present work, four network configurations of CSRNet are proposed with similar configuration but with altered dilation rate. For front-end, a VGG-16 network is adapted, and  $3 \times 3$  kernels are used. As per Kang et al. [10] and Shen et al. [11], network with small kernels with more layers shows better efficiency than fewer layers with larger kernels. The number of VGG-16 layers must remain constant once fully connected layers are dropped. The most important consideration is weighing the accuracy versus resource overhead, which includes training time, memory usage, and



**Fig. 3** Comparison of maxpooling, convolution, and upsampling

parameter count. In order to minimize the detrimental effects of pooling operation on output accuracy, the experiment shows an optimum trade-off that may be reached by limiting the first 10 levels of the VGG-16 to only three pooling layers rather than five. A bilinear interpolation with factor 8 is used because CSRNet's output is smaller and to guarantee that output preserves same resolution as input image. Table 1 shows different convolutional layers.

## 4 Experiments and Results

### 4.1 Dataset Overview

The Shanghai tech dataset comprises a significant crowd count dataset that includes 1198 crowd images. Part-A and Part-B are the two divisions of the dataset where each of them contains 482 and 716 photographs, respectively. The train and test subsets of Part-A include 300 and 182 pictures, respectively. The train and test subsets of Part-B are made up of 400 and 316 pictures, respectively. Images gathered from the Internet are displayed in Part-A, and images taken on Shanghai's crowded streets are displayed in Part-B. Figure 4 depicts images from Shanghai tech dataset.

The training accuracy specifies model's performance on training data, while validation accuracy shows model's performance on new data. Figure 5 shows training data with validation accuracy and loss.

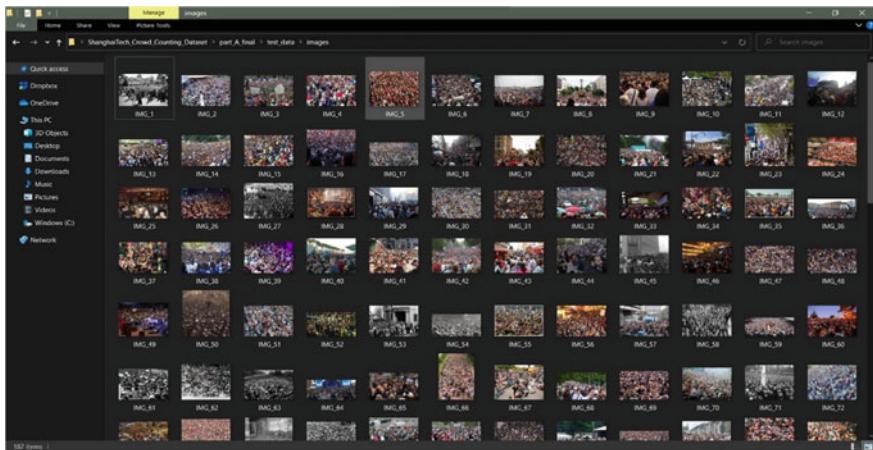
**Table 1** Convolutional layers with kernel size and dilation rate

Configuration of CSRNet			
A	B	C	D
Input (unfix-resolution color image)			
Front-end (fine-tuned from VGG-16)			
conv3-64-1			
conv3-64-1			
Maxpooling			
conv3-128-1			
conv3-128-1			
Maxpooling			
conv3-256-1			
conv3-256-1			
Maxpooling			
conv3-512-1			
conv3-512-1			
conv3-512-1			
Back-end (four different configurations)			
conv3-512-1	conv3-512-2	conv3-512-2	conv3-512-4
conv3-512-1	conv3-512-2	conv3-512-2	conv3-512-4
conv3-512-1	conv3-512-2	conv3-512-2	conv3-512-4
conv3-256-1	conv3-256-2	conv3-256-4	conv3-256-4
conv3-128-1	conv3-128-2	conv3-128-4	conv3-128-4
conv3-64-1	conv3-64-2	conv3-64-4	conv3-64-4
conv1-1-1			

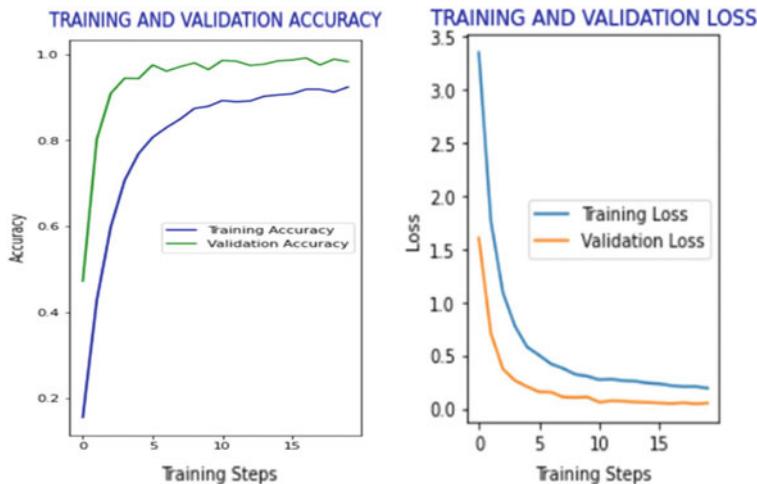
- If the validation loss increases with decreasing accuracy, then the model is cramming values or not learned properly.
- When loss increases while validation accuracy also increases, then the model is inferred as overfitting.
- In validation phase, loss is reduced with the increase in accuracy inferring that model is learning and operation is acceptable.

## 4.2 Performance Analysis

Several approaches detailed in Sect. 2.1 are employed for crowd counting. In the present work, CSRNet is employed as believed to be efficient in predicting the count. To accommodate different head sizes, MCNN employs three columns of CNNs.



**Fig. 4** Images from Shanghai tech dataset



**Fig. 5** Training data with validation accuracy and loss

Moreover, MCNN is observed to show good adaptability. With the use of switch, Switch-CNN utilizes local density variation for classification of images in the crowd. With dilated convolution layers, CSRNet facilitated density detection without reducing spatial resolution. Table 2 presents the comparison of CSRNet with the existing models. It is observed that CSRNet achieves lowest MAE and MSE as shown in the below table. The results indicate that CSRNet is the most compressed and best-performing CNN. Comparison of performance of CSRNet against existing approaches is shown in Table 2.

**Table 2** Comparative analysis of CSRNet with existing approaches referred

Approach	MAE	MSE
Multi-column CNN	181.8	277.7
Fully convolution network	126.5	173.5
MCNN	110.2	173.2
Cascaded-MTL	101.3	152.4
Switching-CNN	90.4	135
<b>CSRNet</b>	<b>68.2</b>	<b>115</b>

## 5 Conclusion

In the present work, crowd counting is implemented using the CSRNet architecture. The dilated convolutional layers are employed to combine multi-scale-related information in the congested images. The benefits of dilated convolutional layers are considered, and CSRNet could be expanded to the receptive field without dropping resolution. This architecture can also be used to count vehicles and used for traffic analysis and anything that requires counting. The proposed approach achieved accuracy of 85.31% on Shanghai tech dataset. This will be applicable to real-world problems like traffic surveillance.

## References

- Boominathan L, Kruthiventi SS, Babu RV (2016) CrowdNet: a deep convolutional network for dense crowd counting. In: Proceedings of the 24th ACM international conference on multimedia. <https://arxiv.org/abs/1608.06197>
- Marsden M, Guinness KM, Little S, O'Connor NE (2016) Fully convolutional crowd counting on highly congested scenes. <https://arxiv.org/abs/1612.00220>
- Sindagi VA et al (2017) CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In: Advanced video and signal based surveillance (AVSS), pp 1–6
- Bhangale J, Patil S, Vishwanath V, Thakker P, Bansod A (2020) Near real-time crowd counting using deep learning approach. In: 3rd International conference on computing and network communications. Procedia Comput Sci 171:770–770
- Zhang Y, Zhou D, Chen S, Gao S, Ma Y (2016) Single-image crowd counting via multi-column convolutional neural network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA, pp 589–597. <https://doi.org/10.1109/CVPR.2016.70>
- Sam DB, Surya S, Babu RV (2017) Switching convolutional neural network for crowd counting. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), Honolulu, HI, USA, pp 4031–4039. <https://doi.org/10.1109/CVPR.2017.429>
- Hu Q, Wang P, Shen C, van den Hengel A, Porikli F (2018) Pushing the limits of deep CNNs for pedestrian detection. IEEE Trans Circuits Syst Video Technol
- Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. <https://arxiv.org/abs/1409.1556>

9. Li Y, Zhang X, Chen D (2018) CSRNet: dilated convolutional neural networks for understanding the highly congested scenes. In: IEEE conference on computer vision and pattern recognition, pp 1091–1100
10. Kang D, Ma Z, Chan AB (2019) Beyond counting: comparisons of density maps for crowd analysis task; counting, detection, and tracking. *IEEE Trans Circ Syst Video Technol*
11. Shen Z, Xu Y, Ni B, Wang M, Hu J, Yang X (2018) Crowd counting via adversarial cross-scale consistency pursuit. In: Proceedings IEEE conference on computer vision pattern recognition

# Evaluation Methods of CDIO Project at Duy Tan University



Van-Truong Truong and Anand Nayyar

**Abstract** Using the CDIO framework in project teaching has been proven to help improve students' teamwork, communication, and critical thinking skills. However, the CDIO class at Duy Tan University (DTU) was at risk of significant fragmentation, as groups working on different projects were not interested in participating in discussions and sharing with other groups. It happened because their knowledge is insufficient to discuss another project in-depth, the time spent in cross-group discussions is short, or simply because of student apathy. Moreover, the Covid-19 pandemic has made implementing learning groups in CDIO much more difficult. Numerous ways have been tried to engage students in learning and adapt to specific conditions before, during, and after the pandemic. Before the pandemic, we increased the enjoyment of learning for students by creating a "fun competition" using DTU-Clicker, the electronic voting device for end-of-class quizzes. The winning team in the poll will receive bonus points, while the other groups will receive the knowledge that the CDIO instructor wants to convey. During Covid-19, we must shift the CDIO projects to an online teaching format. Accordingly, we use Kahoot! to create games in learning CDIO for class members, as well as members of a group. Back after the pandemic, we used Plicker as an enhancement to DTU-Clicker. This paper presents some sharing and comparisons about implementing CDIO teaching at DTU regarding the mentioned techniques. We emphasize the importance of the CDIO instructor's method of imparting knowledge and students' excitement and passion when learning CDIO. At the same time, the results show DTU's efforts in adapting to different teaching conditions.

**Keywords** Clicker · Plicker · CDIO implement · Assessment for learning · CDIO standards: 5, 11

---

V.-T. Truong

Faculty of Electrical-Electronic Engineering, Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam  
e-mail: [truongvantruong@dtu.edu.vn](mailto:truongvantruong@dtu.edu.vn)

A. Nayyar (✉)

School of Computer Science, Faculty of Information Technology, Duy Tan University, Da Nang 550000, Vietnam  
e-mail: [anandnayyar@duytan.edu.vn](mailto:anandnayyar@duytan.edu.vn)

## 1 Introduction

The CDIO framework has been deployed in nearly 200 universities worldwide in most geographic regions such as Europe, North America, Asia, Latin America, UK–Ireland, Australia, New Zealand, and Africa. The fields of the practical application of CDIO include university aerospace, applied physics, electrical engineering, and mechanical engineering departments [2]. The implementation of CDIO-oriented teaching and learning has proven to bring many advantages. First and foremost, CDIO helps to align the labor market's needs with the outcome quality, helping students bridge the gap from the school to the natural working environment [3]. Moreover, CDIO helps learners have the opportunity to develop comprehensively, being able to adapt to the changes of the times and the working environment. On the school side, the implementation of CDIO creates a modern, effective, and convenient educational standard for reviewing and adjusting training programs to suit social realities and recruits' needs.

However, achieving the highest efficiency in each CDIO classroom is always challenging for every school. In addition to human limitations such as the small number and quality of good CDIO instructors, the varying level of students in the class, or infrastructure limitations, many other factors still affect the effectiveness of CDIO classes. From the perspective of almost every group of students implementing a CDIO project, because their topic orientation is obvious and can be independent of other groups, there will appear cases where groups of students cannot interact well. Although each group could complete their CDIO project well, the breakdown in cross-talk made the CDIO class less engaging. It happens because groups of students do not keep up with the work progress of other groups or do not have the need to learn more about other topics. As a result, cross-talk sessions risk becoming one-on-one conversations between the CDIO instructor and each group.

Besides, in the past three years, the Covid pandemic appeared and broke out, causing adverse effects on education. Most countries have had to close entirely or partially to limit the epidemic, leading to the immediate closure of educational institutions. In Vietnam, the school year plans of educational institutions were interrupted, and educational programs and contents had to be changed to focus on the core part. Nearly, 20 million students have not been able to go to school for a very long time, leading to the inevitable online training process [6]. The CDIO project subjects at the Faculty of Electrical and Electronic Engineering (FEEE), Duy Tan University (DTU), also had to change to adapt to new conditions but faced many challenges. Specifically, the change in learning conditions raises the following questions:

- How to bring interest in learning CDIO in the context that all communication has to be done through the computer screen (online learning)?
- How to ensure students are paying attention to the instructions from CDIO instructors?
- Are the methods taught during the pandemic still relevant after the new normal is opened?

Those are the questions to which every CDIO instructor and the management board at DTU constantly struggle to find the answer. Of course, it is impossible to perfect a CDIO teaching system that meets all of the above problems in a short time. However, we have tried to devise the most appropriate solutions possible. Accordingly, in this research paper, we present and evaluate some experiences implementing interest-enhancing CDIO learning at DTU before, during, and after the Covid pandemic. These efforts have confirmed the continuous improvement in CDIO teaching at DTU which is considered an essential criterion in ensuring quality output for students.

## Organization of Paper

The rest of the paper is organized as: Section 2 focuses on methodology specifying the period before the COVID-19 pandemic, the period in the COVID-19 pandemic and the period after the COVID-19 pandemic. Section 3 highlights discussion. And, finally Sect. 4 concludes the paper.

## 2 Methodology

### 2.1 *The Period Before the COVID-19 Pandemic*

First, we briefly present the CDIO teaching system at the FEEE, DTU. We have five CDIO project subjects implemented according to the student's knowledge levels from low to high. Each project subject is taught in standard laboratories for 45 h. The project is carried out and evaluated in groups; each group has a maximum of three members. During class hours, half of the time is for students to practice professional skills such as designing circuit boards, learning electrical-electronic components, analyzing schematic diagrams, soldering electronic circuits, operating electronic circuits, embedded programming, building user interfaces, and testing algorithms. The rest of the time is used for presentations, discussions, and cross-talk [8]. In addition to expertise, we consider these activities to be essential process criteria for grading each student. We build rubric sets with specific criteria for assessing problem-solving skills, writing skills, presentation skills, and technical skills. This traditional CDIO teaching and learning process took place in 2020 and earlier.

During this period, to increase interest in CDIO learning and improve the “positive” atmosphere in the classroom, we implemented the DTU-Clicker system. It is a system that combines hardware and software, supporting the teaching process as follows:

- The CDIO instructor asks questions prepared by PowerPoint through the projector screen, as shown in Fig. 1.
- Each student/group of students answers questions via a hand-held wireless communication device, what we call the DTU-Sender, as shown in Fig. 2. Each DTU-Sender has a unique ID.



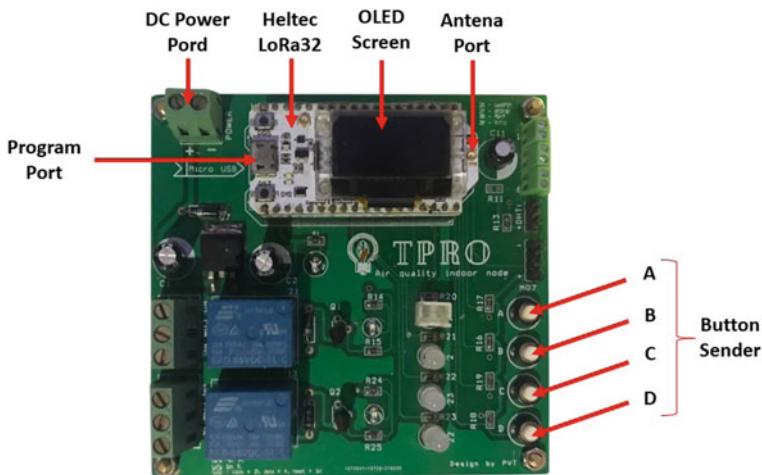
**Fig. 1** A CDIO session at the Electronic Lab, Hoa Khanh Nam Campus, DTU and their result

- A receiver, called a DTU-Receiver, as shown in Fig. 3, is connected to the CDIO Instructor’s computer to aggregate questions and produce specific statistics. The information in each recorded answer includes the student’s election result, the time of submission, and the DTU-Sender ID.
- The CDIO instructor returns the correct answer to each group by the reverse path from the computer software sent through the DTU-Receiver to the DTU-Sender. If the student’s answer is correct, the LED on the DTU-Sender will light up accordingly.

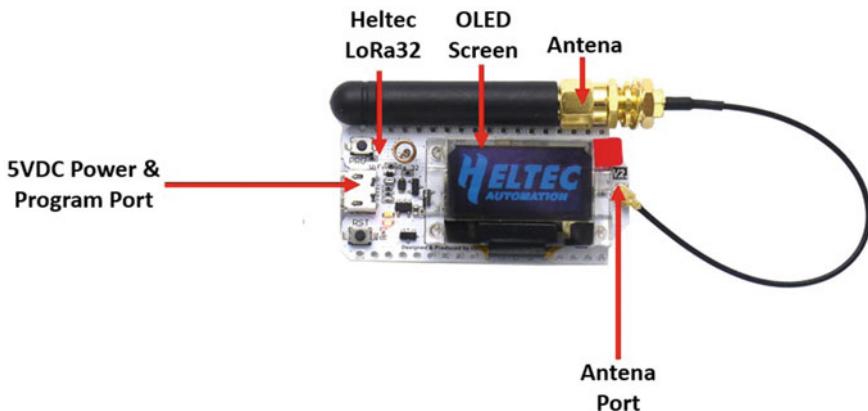
A novel system has been designed by keep the idea of Wentao et al. [10] and also added some more advanced features. All the open source reference materials for Hardware and Software Design is available at: [https://drive.google.com/drive/folders/1q5QKe\\_QMneDk0iT-Pnv8yfEpM3nFypV?usp=drive\\_link](https://drive.google.com/drive/folders/1q5QKe_QMneDk0iT-Pnv8yfEpM3nFypV?usp=drive_link).

Many CDIO classroom activities with various question types can be applied with DTU-Clicker. Each CDIO instructor, depending on his teaching style, can customize how to use this system, but all have the same goal of enhancing group communication, increasing focus on lessons, and improving student learning interest. Some of the classroom activities are enlisted as follows:

- Attendance: Instead of using the traditional and boring name-feedback method, we use the DTU-Clicker to take attendance. Usually, a straightforward mnemonic multiple-choice question is used for roll call. Of course, there is no “wrong answer” in this case because the ultimate goal is to collect information about students studying in class, i.e., using DTU-Clicker, quickly and effectively.
- Increases concentration: Sometimes, during the instruction of a professional operation, it is difficult to confirm whether students are following and understanding the lesson, especially in the technical field. To solve this problem, we often use the application form of a fill-in-the-blank question, which requires students to apply



**Fig. 2** Design of DTU-Sender



**Fig. 3** Design of DTU-Receiver

their knowledge and understanding to choose an appropriate term/concept/solution for the problem. A plus point is an accompanying bonus for groups or individuals who answer correctly. Not only does this increase concentration, but it also improves the classroom atmosphere significantly if done regularly by the CDIO instructor.

- Discussion: As discussed above, engaging groups of students to cross-talk can strengthen and increase knowledge content very well. Compared with the traditional discussion method, we find that DTU-Clicker can better improve this activity. Usually, many shy students do not dare to express their opinions or do not want to participate in discussions on topics belonging to other groups, so their discussion scores are deficient. With DTU-Clicker, we often use multiple-choice

opinion-based questions: strongly agree, agree, and disagree. For example, the question asked is, “For the topic Designing System A, group X used method T. Do you agree with this solution?” Furthermore, group discussions, as well as cross-groups, take place to clarify the issue. In many cases, better solutions to one group’s problem can come from other group members.

- Summary knowledge: Instead of paper tests, DTU-Clicker creates a “fun competition” at the end of each class. We have prepared ten multiple-choice questions to determine the winner group, the group with the correct answers, and the fastest.
- Besides, DTU-Clicker can also be developed and applied for process evaluation, peer instruction [4], and question-driven instruction [1].

## 2.2 *The Period in the COVID-19 Pandemic*

Starting from the beginning of 2020, the epidemic outbreak led to all classes at DTU having to switch to online form, including CDIO project classes. We use the Zoom and Sakai platforms integrated into the Learning Management System (LMS) MyDTU, specifically designed by us to deploy online classes, as shown in Fig. 4. To ensure the online teaching and learning process goes smoothly, DTU has invested and prepared very carefully regarding facilities and infrastructure. Specifically, 27 servers are operating on the myDTU system, 50 servers operating on the Zoom meeting system, domestic bandwidth connecting the ten megabits/s port directly to the national transit station, and we also invested 150 wide-angle, high-resolution cameras for teaching and online exams for students during the epidemic season.

However, we encountered many challenges when implementing online teaching of CDIO subjects in electrical-electronics. Our DTU-Clicker system that we usually use is not applicable, making classroom implementation very difficult. In the first phase, we use traditional methods to assess students, assigning assignments and requesting submissions through the LMS system. However, we found that this method is not effective because the answer sheets of the groups of students tend to be similar. It happens when weak students seek help from others to improve their grades; this process cannot be monitored in an online environment.

In order to address a problem in teaching CDIO, we utilized a tool called Kahoot [9], which is a free and interactive teaching aid designed for online quizzes. Kahoot is a web-based application that can be accessed on any device with an internet connection, such as laptops, smartphones, or computers. It is also compatible with Zoom as an additional software. The use of game theory in teaching CDIO with Kahoot helps to engage students, inspire them, and decrease their dependence on one another when answering questions. Additionally, Kahoot is a user-friendly software that is lightweight and compatible with multiple web browsers.

To use Kahoot, the CDIO instructor must first create an account and enter prepared questions on the web interface. Students can then access the Kahoot website on their devices and join the game using a game pin and a nickname as their account.

After each question, students will receive immediate feedback on their answers and the fastest answer will also be displayed. The CDIO instructor can save these results for later evaluation or use them to determine the next lesson plan. Kahoot supports various question formats, including quizzes, jumbles, discussions, and surveys.

However, due to limited communication in the online environment, group members cannot discuss or exchange ideas directly on the Zoom channel. Therefore, during this period, Kahoot can only be used for Warm-Up and Knowledge Summarization activities, as reported by [5]. The questions chosen for these activities are focused on the core concepts of CDIO, are simple, and can be answered quickly by students. Here are a few examples:

**1. True/False Questions:** These are simple questions with two possible answers—true or false. They are easy to answer and can test students' understanding of key concepts in CDIO.

Example: True or False: The “O” in CDIO stands for “operate.”

**2. Matching Questions:** These questions require students to match one set of items with another set. They can be used to test students' understanding of definitions, concepts, or terminology.

Example: Match the following CDIO phase with its corresponding activity: – Conceive – Design – Implement – Operate

- A. Building and testing
- B. Planning and defining
- C. Developing and evaluating
- D. Assessing and improving.

**3. Fill-in-the-Blank Questions:** These questions require students to fill in the missing word or phrase in a sentence. They can be used to test students' ability to recall specific information or terminology.

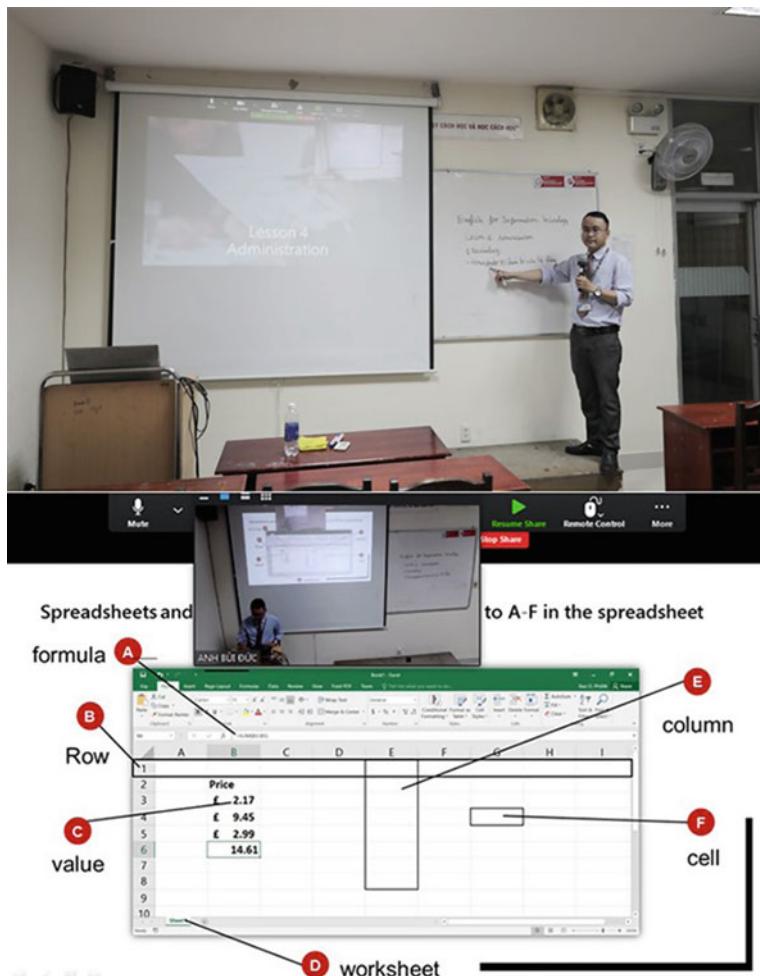
Example: The “D” in CDIO stands for “ ..... and evaluating.”

**4. Multiple-Answer Questions:** These questions have more than one correct answer. They can be used to test students' ability to identify multiple aspects of a concept or topic.

Example: Which of the following are key components of the CDIO framework?

- A. Conceive
- B. Design
- C. Implement
- D. Operate
- E. Evaluate

Overall, the type of question used in a Warm-Up activity depends on the learning objectives and the level of difficulty appropriate for the class. Kahoot provides a range of question formats, making it easy for instructors to choose the best format for their Warm-Up activity.

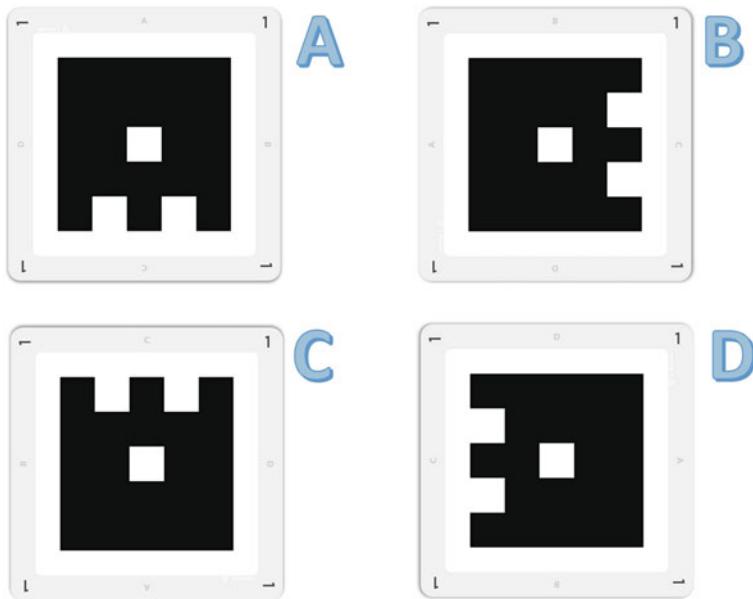


**Fig. 4** DTU lecturer was implementing an online class

### 2.3 The Period After the COVID-19 Pandemic

After the pandemic at the beginning of 2022, we returned to on-site teaching. Besides DTU-Clicker and Kahoot, we also implemented another application as an upgrade and fallback, named Plicker [7]. Plicker, also known as “paper clicker,” is a tool to help organize the review and multiple-choice tests in the CDIO classroom in an effective and fun way. Furthermore, the requirements to conduct classroom teaching by this method are also elementary:

- CDIO instructors are equipped with smartphones.
- Classrooms have computers connected to the Internet.



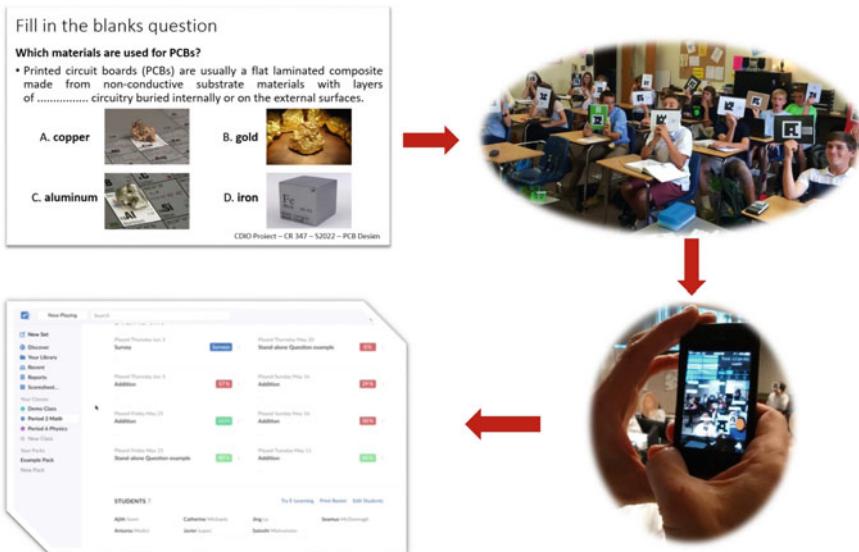
**Fig. 5** An example of a Plicker card

- Each student is given a bar code printed card, as shown in Fig. 5.

It significantly reduces costs compared to the DTU-Clicker system if class sizes increase. The process of implementing Plicker in CDIO teaching includes four stages, as depicted in Fig. 6:

- **Step 1:** The CDIO instructor prepares the Plicker cards and distributes one to each student. Each Plicker card is encrypted with a different bar code; the four edges of the card correspond to four answers A, B, C, and D. The set of questions is also prepared on PowerPoint software.
- **Step 2:** Students read the question on the screen, choose the correct answer, and then bring the edge with the corresponding option up.
- **Step 3:** The CDIO instructor uses the “camera” icon on the phone screen and swipes the phone over the student’s card to collect responses.
- **Step 4:** Plicker software collects images of cards and records the answers. Accordingly, the CDIO instructor can know the data of the answers that the students have chosen, the statistics of which students answered correctly, and convert it to % of the score achieved by each student. In addition to the summary table in the class list order, there is also a result table according to the ranking from high to low, so it becomes effortless for CDIO instructors to give bonus points.

Another significant advantage we realize when using Plicker is the ability to correctly calculate the percentage of students who answer the question. Combined with a set of question banks in which each question is associated with CDIO’s outcome



**Fig. 6** Illustrating the four-stage process of implementing Plicker in teaching CDIO

standards, we can quickly grasp the results of students' knowledge to propose appropriate teaching directions. Additionally, test data is automatically saved per student at the Plickers website so CDIO instructors can monitor and assess student progress.

### 3 Discussion

In the above section, we have presented some implementations of CDIO teaching at FEEE, DTU through stages. Each method has its advantages and disadvantages, depending on the ability and teaching orientation of each CDIO instructor, which is appropriate. The common point in all three proposed methods is that they are all technical solutions. Therefore, when using, users must be familiar with the implementation method and can handle situations well because technical errors can appear at any time. In other words, backup situations are necessary to prepare in case the proposed methods appear to fail.

Another point to note is that implementing these methods is quite time-consuming. The first is that the CDIO instructor's lecture preparation time must be significantly increased to serve the question banks and attractive case studies. This process can only be perfected through continuous updating and improvement over many classes. Next, implementing them in the classroom requires preparation time to distribute DTU-Sender or Plicker cards to students. As the "game" progresses, the CDIO instructor also needs time to control the emotions of the classroom so that it does not exceed the allowable threshold.

**Table 1** Comparison of CDIO teaching methods in FEEE, DTU

Criteria	DTU-clicker	Kahoot	Plicker
Investment cost for the system/class size	Large/small	Large/large	Small/large
Operating conditions	On-site	On-site/online	On-site/online
Activities implemented in class	Diversity	Medium	Medium
Support faculty in student assessment	Yes	Yes	Yes
Enhance students' interest and engagement in the CDIO project	Yes	Yes	Yes

Another difficulty when using these methods is the uneven quality of CDIO classes within the same faculty. For example, knowledge area A was quickly absorbed by the group of students in class CDIO 1, so the instructor immediately turned to new topics. However, it is still an area of knowledge A, but students of class CDIO 2 cannot understand, as shown by deplorable statistics. It leads to the requirement that either the CDIO instructor must take care of all contingencies or they must change the teaching plan to help students ensure a standard of output.

Some specific pros and cons are listed below:

*About the pros:*

- All three solutions help increase student interest in learning and engagement with CDIO subjects and support CDIO instructors' assessment well.
- DTU-Clicker is a self-developed product of FEEE, DTU, so it is possible to expand the features.
- Kahoot and Plicker are easy to implement in practice due to their enormous resources and user community.
- Clicker can be used in classes with many students, up to 60 people.

*About the cons:*

- DTU-Clicker is challenging to deploy on a large scale due to hardware limitations. During the same period, only hardware-equipped CDIO classes could implement this approach.
- Kahoot and Plicker are relatively limited when implementing layer operations compared to DTU-Clicker. These two methods are also manufacturer dependent and provide fixed features.

Table 1 summarizes the comparison of CDIO teaching methods in FEEE, DTU.

## 4 Conclusion

The research paper presents some methods of teaching and evaluating CDIO projects at FEEE and DTU through each stage. Before the pandemic, we used DTU-Clicker, a self-developed product of CDIO instructors at the faculty, to conduct teaching and

student assessment activities. During the pandemic, due to the need to change the form of teaching to online, we use Kahoot software in combination with the Zoom platform to enhance students' engagement with the lesson. Then, when we returned to on-site learning, we diversified the CDIO learning experience for students with Plicker. We aim to increase students' interest in learning and engagement with CDIO subjects in all three methods. Furthermore, through these methods, students improve their ability to communicate, make decisions, and discuss. Instructors have tools that make it easier to deliver their classes and effectively assess students. In each period, FEEE has made timely changes to adapt to new conditions. It has proven our efforts to improve teaching, thereby contributing to improving the quality of student outcomes.

## References

1. Dawar D (2023) Question driven introductory programming instruction: a pilot study. *J Inf Syst Educ* 34(2):231–242
2. Edstrom K, Kolmos A (2014) Pbl and cdio: complementary models for engineering education development. *Eur J Eng Educ* 39(5):539–555
3. Kontio J (2017) Why universities want to join cdio. In: 13th International CDIO conference, calgary, Canada
4. Lan BL, Lim PM, Ho PW (2022) A modified peer instruction versus teacher's instruction. arXiv preprint [arXiv:2201.10804](https://arxiv.org/abs/2201.10804)
5. Martin-Somer M, Moreira J, Casado C (2021) Use of kahoot! to keep students' motivation during online classes in the lockdown period caused by covid 19. *Educat Chem Eng* 36:154–159
6. Pokhrel S, Chhetri R (2021) A literature review on impact of covid-19 pandemic on teaching and learning. *Higher Edu Future* 8(1):133–141
7. Shana ZA, Abd Al Baki S (2020) Using plickers in formative assessment to augment student learning. *Int J Mob Blended Learn (IJMBL)* 12(2):57–76
8. Truong TV, Ha B, Le B (2019) The effects of industry 4.0 on teaching and learning cdio project at duy tan university. In: Proceedings of the 15th international CDIO conference, p 15
9. Wang AI, Tahir R (2020) The effect of using kahoot! for learning-a literature review. *Comput Educ* 149:103818
10. Wentao C, Jinyu Z, Zhonggen Y (2017) Advantages and disadvantages of clicker use in education. *Int J Inf Commun Technol Educ (IJCCTE)* 13(1):61–71

# Deciphering Stem Cell Pluripotency Using a Machine Learning Clustering Approach



Nikhil Jain, Payal Gupta, Abhishek Sengupta, Ankur Chaurasia, and Priyanka Narad

**Abstract** Complex biological systems including genes, transcription factors, regulators, and signaling pathways control pluripotency. Based on their capacity for self-renewal, pluripotent stem cells can be divided into two categories: naive and primed. Rather than being static states, these stages depict a spectrum. Post-implantation embryonic stem cells that have undergone priming have little capacity for differentiation. This study uses gene expression data from unprimed and primed pluripotent embryonic stem cells to propose a novel clustering approach based on machine learning to evaluate stem cell pluripotency. Unique clusters of genes with varied degrees of pluripotency are found and verified by examining publicly accessible gene expression data and carrying out functional enrichment studies. Gene ontology (GO) analysis, which finds enhanced biological processes, molecular functions, and cellular components associated with gene sets, necessitates specialized tools like clusterProfiler for gene expression analysis. It aids in comprehending how particular circumstances can influence certain processes. Another key component is network analysis, and Cytoscape aids in visualizing and analyzing intricate gene/protein networks.

**Keywords** Stem cells · Gene ontology (GO) analysis · Network analysis · Machine learning

## 1 Introduction

Naïve and primed pluripotency are two different pluripotent states in embryonic stem cells (ESCs) characterized by molecular markers, developmental potential, and response to differentiation markers. Undifferentiated cells called naïve stem cells can develop into any form of cell in the body. They have the capacity to develop into a variety of cell types and are viewed as naïve because they have not yet decided to become a particular cell type [1].

---

N. Jain · P. Gupta · A. Sengupta · A. Chaurasia · P. Narad (✉)  
Amity Institute of Biotechnology, Amity University Uttar Pradesh, Noida, Uttar Pradesh, India  
e-mail: [pnarad@amity.edu](mailto:pnarad@amity.edu)

Mouse embryonic stem cells (mESCs) have clarified naive and primed pluripotency differences, yet this understanding's applicability to human ESCs (hESCs) remained unclear. Transcriptomic analysis of hESCs revealed two pluripotent substates, termed hESC1 and hESC2. hESC1 resembled naive mESCs, exhibiting higher expression of pluripotency genes (e.g., NANOG, KLF4, TDGF1), plus cell cycle and DNA replication genes. Conversely, hESC2 paralleled primed mESCs, showing increased expression of lineage-specification genes (e.g., T, MIXL1, EOMES). Gene ontology analysis highlighted hESC1's cell cycle and DNA-related terms, while hESC2 related to development and differentiation. Signaling pathways differed, with hESC1 activating JAK/STAT and hESC2 showing higher TGF $\beta$ /Activin A and FGF2 activation [2–7].

Through GO analysis, gene expression analysis makes use of tools like clusterProfiler to comprehend biological processes [8, 9]. In order to identify changed processes in gene groups, clusterProfiler searches for enriched GO keywords. In this investigation, networks made possible by Cytoscape demonstrate links between genes and proteins. Network analysis is improved with plugins like MCODE, BINGO, and JEPETTO [10, 11]. Patterns are found using dimensionality reduction in machine learning techniques like PCA. Genes are grouped by expression in K-means clustering, whereas co-regulated genes are grouped in dendrogram analysis. These techniques allow for thorough gene expression analysis.

## 2 Methodology

### 2.1 Collection of Data

The data was extracted from the Gene Expression Omnibus (GEO) database maintained by the National Centre for Biotechnology Information (NCBI) (<https://www.ncbi.nlm.nih.gov/geo/>). The dataset selection criteria focused on RNA sequencing gene expression data (RNA-seq) with ample samples, excluding methylation and medication effects and requiring .csv or .txt format. The chosen dataset, GSE208300, comprises 60 samples representing primed and naive pluripotent stem cells, offering distinct molecular insights into pluripotent stages via computational and statistical analyses. This investigation unveils regulatory mechanisms, physiological functions, and disparities between these pluripotent states [11, 12].

### 2.2 Preprocessing of Data and Analysis

The preprocessing of the RNA-seq data includes several steps, and these steps were executed using R programming language, and the packages like GEOquery, DESeq2, and dplyr were implemented. Some of the major steps in preprocessing were: (1)

**Normalization:** The read counts are normalized to account for differences in library size using tools such as DESeq2. (ii) **Differential gene expression analysis:** Finally, differential gene expression analysis is performed using the tool DESeq2 to identify genes that are differentially expressed between the experimental groups [13].

### 2.3 Building Accuracy of the Data

N.A. values in gene expression data can lead to inaccuracies in statistical analysis and affect the identification of differentially expressed genes. Duplicate values can also affect the integrity of the data, as they can bias the analysis and lead to false positives. Thus, N.A. and duplicate values are removed from the normalized data. The samples to be used for gene ontology are then filtered in the R programming language itself. The genes with  $\log FC > 2.0$  and  $p\text{-value} < 0.01$  were considered up-regulated and overexpressed in pluripotency, while those with  $\log FC < -2.0$  and  $p\text{-value} < 0.01$  were considered down-regulated and under-expressed in the pluripotency [14].

### 2.4 Model for Machine Learning

The unsupervised clustering model was developed using Python language. For this, packages such as NumPy, pandas, matplotlib, pyplot, seaborn, etc., were imported into the environment. For visualization, different types of plots were included including PCA cluster plot, dendrogram, hierarchical clustering dendrogram, Elbow curve, and crescent curves using K-means value. Since all 16,381 genes will not give significant results, hence, genes with variance  $< 0.3$  were dropped from the data.

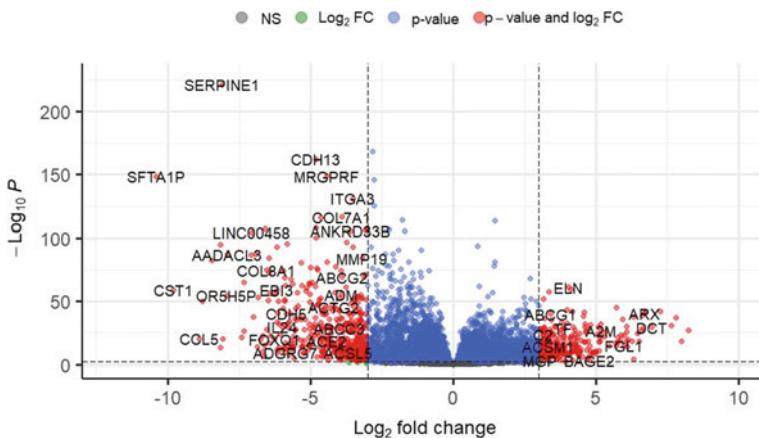
## 3 Results and Discussion

### 3.1 Identification of DEGs

The DEGs were identified with a threshold of  $|logFC| > 2$  and  $p\text{-value} < 0.05$  resulting in 243 up-regulated and 432 down-regulated DEGs as shown in Fig. 1.

### 3.2 Variables and Clusters

By converting the original variables into a new set of uncorrelated variables called principal components, PCA can identify patterns in a dataset. The expression of



**Fig. 1** Differential expression plot. Volcano plot of the 675 DEGs. The transverse axis represents the  $\log_2\text{FC}$ , while the vertical axis represents the  $-\log_{10}P$ . Red to the right of the zero: up-regulated genes with a  $\log\text{FC} > 2$ ; red to the left of the zero: down-regulated genes with a  $\log\text{FC} < -2$ ; DEG, differentially expressed gene; FC, fold change;  $P$ ,  $p$ -value

two clusters of up-regulated and down-regulated genes that identified the naive and primed conditions was used by hierarchical clustering to separate the pluripotent stem cell samples.

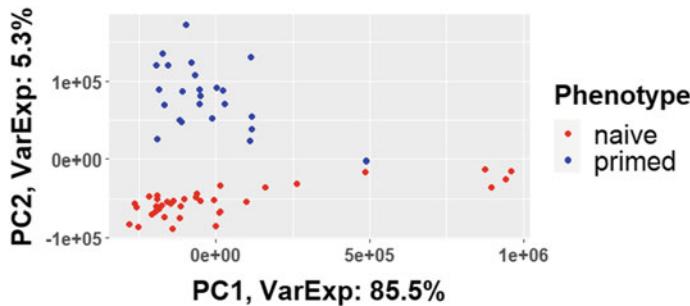
PCA plot of the 60 samples significantly expressed the genes of the stem cells obtained from NCBI. Red color depicts the naïve pluripotent stem cells, and blue color depicts primed pluripotent stem cells.

The majority of the naive samples were densely clustered together, mostly as a result of the substantially higher expression of down-regulated genes between 0 and 0.00001 axis, although the clusters are twisted beyond 0 of the  $x$ -axis (Fig. 2) [15]. The stem cell samples were then divided into groups based on the transition from one state to another by elevated genes. Genes with differential expression were subjected to principal component analysis (PCA), which supported the hierarchical clustering's findings. PCA revealed a distinct cluster segregating the high-impact group away from the low-impact group [16].

#### GO and KEGG Enrichment Analysis.

A common bioinformatics technique for annotating gene function is Gene Ontology (GO). A set of genes-enriched GO terms are visually represented in dot plots using the *R* tool clusterProfiler. This makes it easier to understand how the genes under study relate to molecular functions (MF), biological processes (BP), cellular components (CC), and KEGG enrichment.

GO and KEGG pathway enrichment analysis was used to identify the characteristics of up-regulated and down-regulated DEGs. DNA-binding transcription repressor/activator activity was the major MFs of the 243 elevated DEGs. These genes were mostly implicated in the BPs of forebrain development. The major CC in which these genes were found was synaptic membrane. KEGG enrichment showed that



**Fig. 2** Principal component analysis (PCA) plot

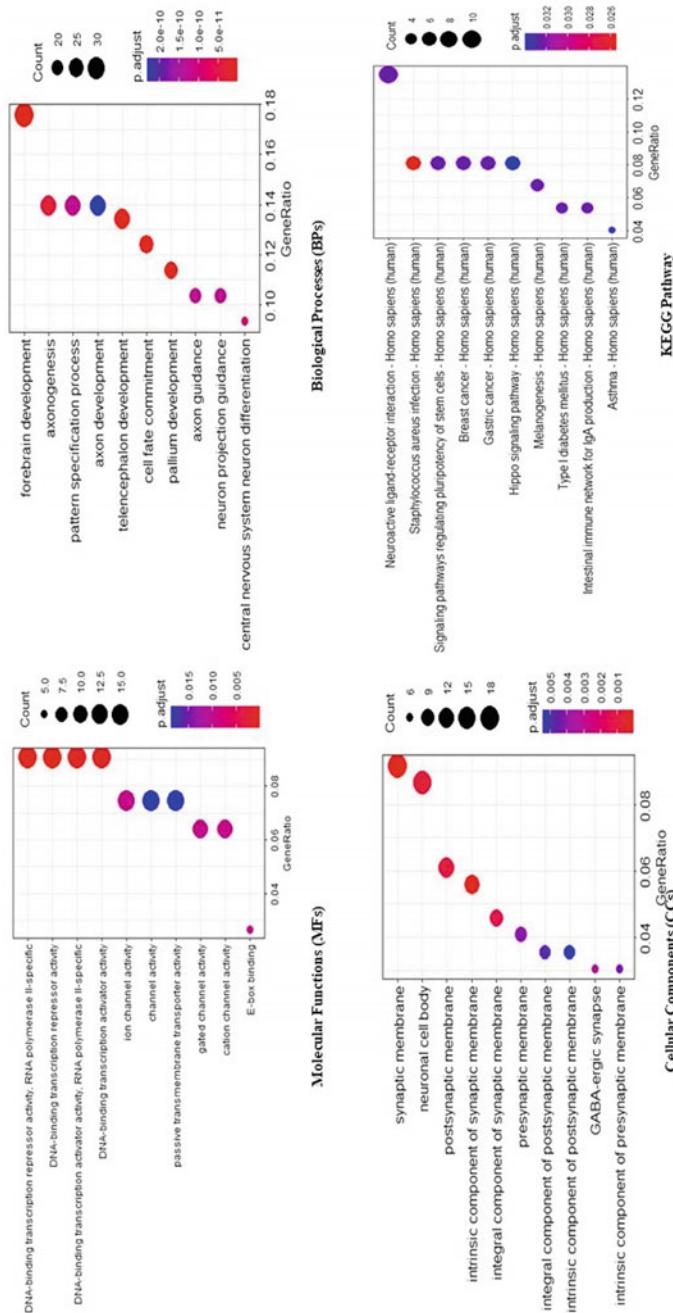
these elevated DEGs are primarily connected to certain pathways, including those governing the pluripotency of stem cells and neuroactive ligand-receptor interaction (Fig. 3).

The collagen containing extra-cellular matrix CCs was the key area of involvement for the 432 down-regulated DEGs. Receptor ligand activity was the main MFs. The down-regulated genes were primarily involved in BPs such as development of the epidermis and the formation of muscle tissue. These DEGs were primarily shown to be involved in pathways like neuroactive ligand-receptor interaction and cytokine-cytokine receptor interaction (Fig. 4).

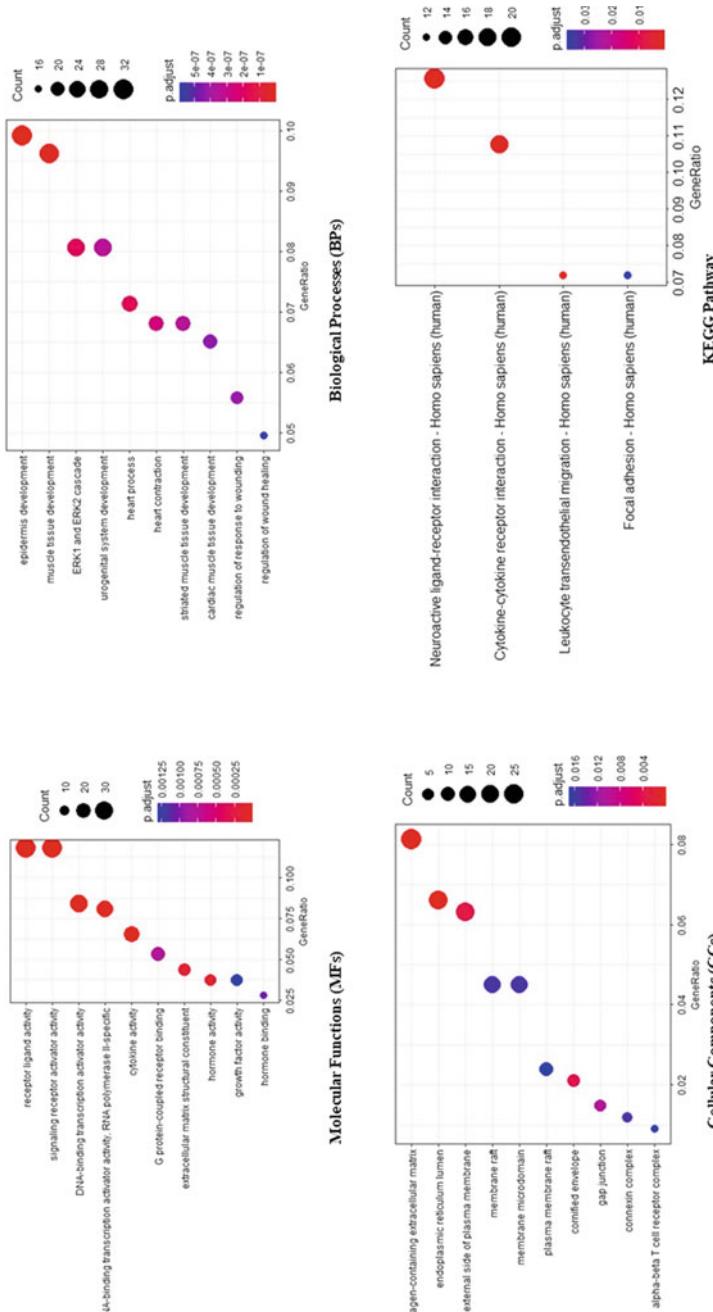
### 3.3 PPI Network Analysis

A PPI network with 267 nodes and 565 edges was constructed to further investigate interactions among the proteins encoded by the DEGs. The MCODE plugin was used to further analyze 175 nodes in order to find hub modules. MCODE assigned a score of 8.923 to the following 12 important genes: CXCL10, EGR1, GBP2, IFI44, IFIH1, IFIT1, IFIT2, IFIT3, ISG15, OAS1, OAS2, and OASL (Fig. 5).

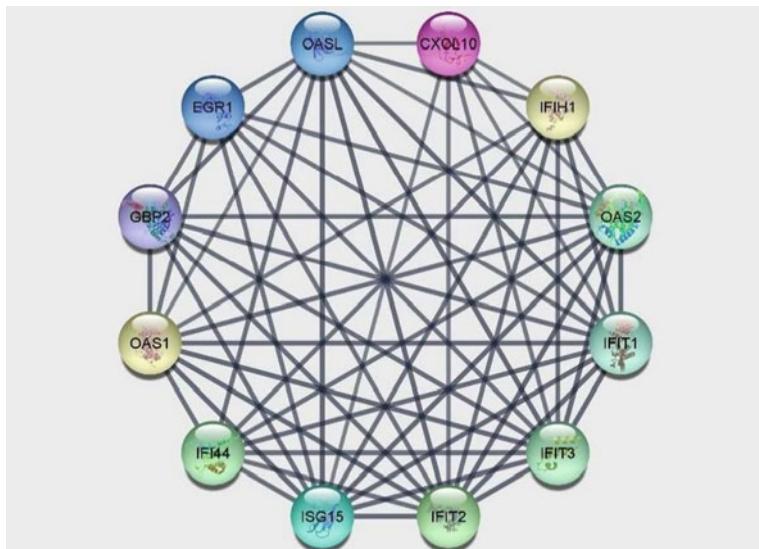
The major pathways and cellular elements that these important genes were engaged in were determined using the BiNGO plugin in Cytoscape. Response to biotic stimuli, immunological response, and cellular reaction to heparin were the main biological activities identified [17]. These genes were associated with the endoplasmic reticulum's integrin complex and the plasma membrane's exterior side. These genes were involved in the molecular processes of DNA binding, 2'-5' oligoadenylate synthetase activity, and nucleotidyl transferase activity. [18–20].



**Fig. 3** Dot plot for Gene Ontology (GO) and KEGG pathway. Dot plots showing Gene Ontology and KEGG analysis results of up-regulated genes with a  $\log_{10}FC > 2$ . The color of each dot represents the  $p$ -adjust value of each term involved in the analysis. The size of each dot represents the gene counts of this term involved in the analysis. BP, biological process; CC, cellular component; MF, molecular function



**Fig. 4** Dot plot for Gene Ontology (GO) and KEGG pathway. Dot plots showing Gene Ontology and KEGG analysis results of down-regulated genes with a  $\log FC < -2$ . The color of each dot represents the  $p$ -adjust value of each term involved in the analysis. The size of each dot represents the gene counts of this term involved in the analysis. BP, biological process; CC, cellular component; MF, molecular function



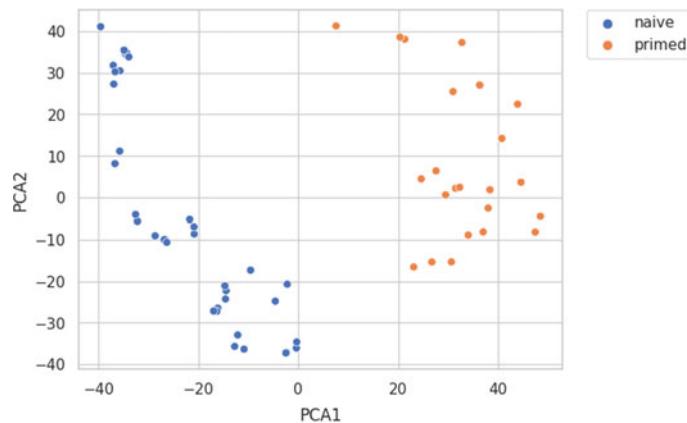
**Fig. 5** Key genes involved in cell functions

### 3.4 Statistical Analysis and Clustering Model Using Machine Learning (ML)

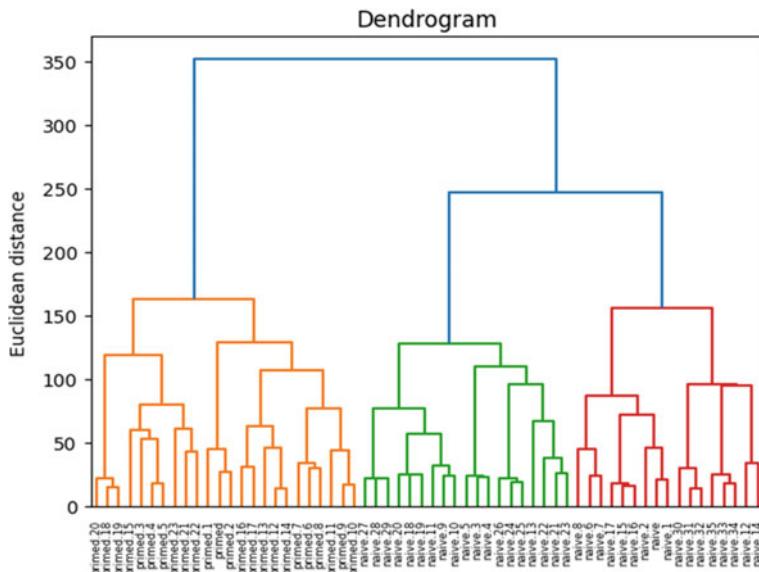
The groups that have been discovered through the application of K-means clustering to the nonzero data are represented by the colors on the PCA plot. A machine learning technique called K-means clustering divides data into discrete groups based on how similar they are, with each group being given a different color on the plot. As a result, the various colors on the PCA map represent the clusters or groups of data points that the K-means clustering procedure has managed to identify (Fig. 7).

The K-means clustering groups the type of cells, i.e., naïve and primed hESCs into clusters. The Euclidean distance depicts the distance between the groups of clusters formed, and the clusters are formed between the same types of cells as they carry similar functions. As seen in Fig. 6, the PCA plot shows the presence of clusters of naïve as well as primed cells, but there is a presence of a gap between the clusters. In Fig. 8, the range of 150–200 Euclidean distance shows the maximum number of significant clusters, i.e., 3 clusters hence,  $K = 3$ .

The dendrogram's X-axis shows the individual data points or clusters of data points that are being grouped. The points are ordered according to their index or the quantity of points they represent, in ascending order. The distance between the data points or clusters of points is shown on the y-axis. The height of each merge on the y-axis represents the distance between the clusters being merged. Thus, the naïve cells GSM60340411, GSM60340412, and GSM60340413 form the closest clusters with minimum distance between them, while primed cells, i.e., GSM6340369, GSM6340370, and GSM6340372 form the closest clusters (Fig. 8).

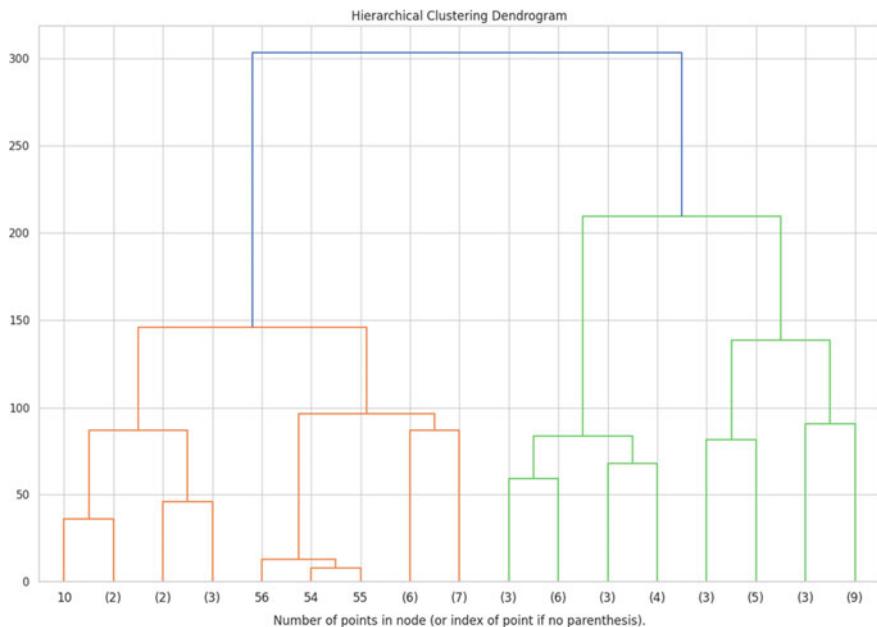


**Fig. 6 PCA cluster plot:** Visualization of data clusters using PCA. Reduced dimensions highlight distinct groupings and patterns within the dataset



**Fig. 7 K-means clustering dendrogram:** hierarchical structure of clusters depicted in a K-means clustering dendrogram. Optimal cluster identification through branch height

The overall distance between each data point and its cluster centroid is quantified by the SSE. It represents the sum of squared Euclidean distances and quantifies the within-cluster variability. Finding the right number of clusters that strike a balance between separation and compactness is the main objective of clustering, which aims



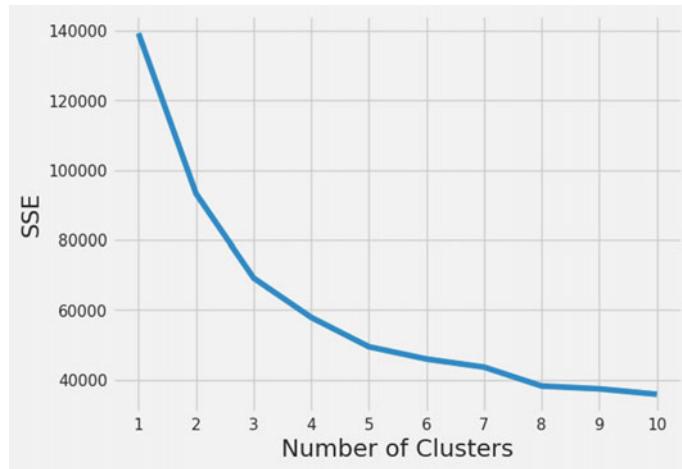
**Fig. 8** Hierarchical clustering dendrogram: based on similarities, the hierarchical clustering dendrogram shows sample relationships. Branches show the degree of dissimilarity, allowing the dataset's hidden patterns and distinctive groups to be found

to reduce this SSE. The SSE and the number of clusters are commonly displayed on the elbow plot's y and x-axis, respectively.

Since smaller clusters can more precisely fit the data points, the SSE tends to decline as the number of clusters rises. However, overfitting and a slight reduction in SSE can result from the addition of too many clusters. The name of the plot comes from the way an elbow is bent. The elbow is marked by this point of inflection, which makes it easy to determine how many clusters are ideal. The elbow plot can be used to help analysts decide how many clusters to utilize, which will improve their comprehension, interpretation, and application of the underlying data structure.

We may deduce that the ideal number of clusters for the given dataset of naive and primed cells is most likely three. According to these traits and similarities, the cells can be successfully divided into three separate clusters. A lower SSE suggests that the data points inside a cluster are more comparable to one another. The SSE of 70,000 reveals the level of diversity within each cluster. With an SSE of 70,000, we may therefore infer that the clustering is largely successful in putting comparable cells in each cluster together (Fig. 9).

The silhouette coefficient's decrease suggests that three clusters are not optimal for naive and primed cells. A coefficient rises to 0.28 and supports 5 clusters as better, enhancing cell isolation within clusters. Bioinformatics techniques, including silhouette curve, elbow plot, dendrograms, and PCA plots, illuminate stem cell pluripotency



**Fig. 9** Elbow plot

disparities. These methods aid in understanding naive and primed cell differences (Fig. 10) [23].

Bioinformatics techniques are essential for analyzing stem cell pluripotency because they shed light on the differences and similarities between naive and primed cells. The silhouette curve, elbow cluster plot, hierarchical clustering dendrogram, K-means clustering dendrogram, and PCA plot are a few methods that are frequently used to accomplish this.



**Fig. 10** Silhouette coefficient curve

The effectiveness of clustering, particularly in distinguishing pluripotent states (naive and primed), is assessed using the silhouette curve, comparing data point similarities within clusters. The optimal cluster count is determined using K-means clustering and elbow plots, ensuring meaningful representation of pluripotent states. Hierarchical clustering dendrograms reveal hierarchical organization, while K-means dendrograms visually separate naive and primed cells. Principal component analysis reduces high-dimensional data to a PCA plot, aiding differentiation between gene expression profiles of naive and primed cells. Integration of these tools helps uncover genes and pathways related to pluripotency, vital for tailored treatments and regenerative medicine. These bioinformatics methods advance stem cell research toward medicinal breakthroughs, enhancing our understanding of pluripotency's potential [8–10].

## 4 Discussion

Naïve and primed are crucial stages of hESCs development and pluripotency. Comprehension of the major genes involved can provide us the opportunity to build biomedical applications of pluripotent stem cells. The key genes obtained by the analysis performed in our study are, namely CXCL10, EGR1, GBP2, IFI44, IFIH1, IFIT1, IFIT2, IFIT3, ISG15, OAS1, OAS2, and OASL, which were highly expressed and found to be linked to pluripotency. These hub genes can help to develop a better understanding of human developmental processes and their application in bioengineering and biomedical research [21, 22]. The potential impact of the hub genes identified can be understood through their roles.

In inflammatory and viral contexts, the chemokine CXCL10 orchestrates immune cell activation and recruitment. Immediate early gene EGR1 responds to stress and viral infections by regulating target genes. GBP2, a guanylate-binding protein, curtails viral replication and spread. Interferon-induced IFI44 curbs viral replication and modulates immune responses. IFIH1 (MDA5) detects viral RNA, triggering antiviral immune reactions. IFIT1, IFIT2, and IFIT3 hinder viral replication via RNA interference. ISG15, an interferon-stimulated gene, aids immune control, protein modification, and antiviral defense. OAS1, OAS2, and OASL genes induce oligoadenylate synthetases activating RNase L to degrade viral RNA and curbing viral propagation.

Machine learning-based clustering identified five distinct stem cell clusters, enhancing understanding of their behaviors and properties. These findings hold promise for regenerative medicine, illuminating molecular processes that govern cellular differentiation, potentially yielding novel disease treatments. Further research is needed to validate and elucidate key genes and pathways in hESC differentiation. Incorporating more datasets and advanced bioinformatics tools could deepen comprehension. Nonetheless, this study significantly advances regenerative medicine and stem cell biology, offering potential for innovative disease therapies [23].

## 5 Conclusion

Our study thoroughly analyzed GSE208300 using bioinformatics tools including clusterProfiler, Cytoscape, and machine learning methods (PCA, K-means, hierarchical dendrogram). Network analysis discovered interrelated genes and modules, furthering our understanding of cellular dynamics, whereas GO analysis revealed hESC growth pathways. Our knowledge of the molecular mechanisms underpinning hESC differentiation has improved as a result of plugins like BiNGO. Important genes like CXCL10, EGR1, and GBP2 were connected to DNA binding and oligoadenylylate synthetase pathways. These markers improve understanding of pluripotency. Our work creates a useful framework for investigating naive and primed cells, processing genes, and illuminating the outcomes [23].

## 6 Future Aspects

The GSE208300-based study comparing naive and primed pluripotent stem cells, utilizing GO analysis, PPI networks, and clustering, suggests new research avenues. Validating results with independent datasets, exploring omics data integration, studying cell state-specific pathways, investigating epigenetic influences, and functionally validating key genes/pathways are potential directions [23]. The research also inspires novel computational methods for accurate cell state prediction. Investigating dynamic gene expression changes during naive-to-primed transition and understanding environmental impacts on state shifts can illuminate regulatory mechanisms. This work could enhance stem cell applications in regenerative medicine and disease modeling by controlling cell fate decisions [23].

## References

1. Narad P, Upadhyaya K, Som A (2017) Reconstruction, visualization and explorative analysis of human pluripotency network. *Netw Biol* 7:57–75
2. Tesar P, Chenoweth J, Brook F, Davies T, Evans E, Mack D, Gardner R, McKay R (2007) New cell lines from mouse epiblast share defining features with human embryonic stem cells. *Nature* 448(7150):196–199
3. Takahashi S, Kobayashi S, Hiratani I (2018) Epigenetic differences between naïve and primed pluripotent stem cells. *Cell Mol Life Sci* 75(7):1191–1203
4. Wang J, Levasseur DN, Orkin SH (2008) Requirement of Nanog dimerization for stem cell self-renewal and pluripotency. *Proc Natl Acad Sci U S A* 105(17):6326–6331
5. Loh Y, Wu Q, Chew J, Vega V, Zhang W, Chen X, Bourque G, George J, Leong B, Liu J, Wong K, Sung K, Lee C, Zhao X, Chiu K, Lipovich L, Kuznetsov V, Robson P, Stanton L, Wei C, Ruan Y, Lim B, Ng H (2006) The Oct4 and Nanog transcription network regulates pluripotency in mouse embryonic stem cells. *Nat Genet* 38(4):431–440

6. Zhang X, Huang C, Chen J, Pankratz M, Xi J, Li J, Yang Y, LaVaute T, Li X, Ayala M, Bondarenko G, Du Z, Jin Y, Golos T, Zhang S (2010) Pax6 Is a human neuroectoderm cell fate determinant. *Cell Stem Cell* 7(1):90–100
7. Ghosh A, Som A (2022) Transcriptomic analysis of human Naïve and primed pluripotent stem. *Cells* 213–237
8. Narad P, Anand L, Gupta R, Sengupta A (2018) Construction of discrete model of human pluripotency in predicting lineage-specific outcomes and targeted knockdowns of essential genes. *Sci Rep* 8(1):11031
9. Narad P (2014) Integrative bioinformatics approaches to analyze molecular events in pluripotency, biology and medicine, 06(03)
10. Shannon P, Markiel A, Ozier O, Baliga N, Wang J, Ramage D, Amin N, Schwikowski B, Ideker T (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* 13(11):2498–2504
11. Ghosh A, Som A (2020) RNA-Seq analysis reveals pluripotency-associated genes and their interaction networks in human embryonic stem cells. *Comput Biol Chem* 85:107239
12. Bhattacharya B, Miura T, Brandenberger R, Mejido J, Luo Y, Yang A, Joshi B, Ginis I, Thies R, Amit M, Lyons I, Condie B, Itskovitz-Eldor J, Rao M, Puri R (2004) Gene expression in human embryonic stem cell lines: unique molecular signature. *Blood* 103(8):2956–2964
13. Subramanian A, Tamayo P, Mootha V, Mukherjee S, Ebert B, Gillette M, Paulovich A, Pomeroy S, Golub T, Lander E, Mesirov J (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci* 102(43):15545–15550
14. Kim J, Chu J, Shen X, Wang J, Orkin SH (2008) An extended transcriptional network for pluripotency of embryonic stem cells. *Cell* 132(6):1049–1061
15. Boyer L, Lee T, Cole M, Johnstone S, Levine S, Zucker J, Guenther M, Kumar R, Murray H, Jenner R, Gifford D, Melton D, Jaenisch R, Young R (2005) Core transcriptional regulatory circuitry in human embryonic stem cells. *Cell* 122(6):947–956
16. Srivastava D, DeWitt N (2016) In vivo cellular reprogramming: the next generation. *Cell* 166(6):1386–1396
17. Hartman K, Bortner J, Falk G, Yu J, Martín M, Rustgi A, Lynch J (2013) Modeling inflammation and oxidative stress in gastrointestinal disease development using novel organotypic culture systems. *Stem Cell Res Ther* 4(S1):S5
18. Bharti S, Sengupta A, Chugh P, Narad P (2022) PluriMetNet: A dynamic electronic model decrypting the metabolic variations in human embryonic stem cells (hESCs) at fluctuating oxygen concentrations. *J Biomol Struct Dyn* 40(10):4570–4578
19. Chen X, Xu H, Yuan P, Fang F, Huss M, Vega V, Wong E, Orlov Y, Zhang W, Jiang J, Loh Y, Yeo H, Yeo Z, Narang V, Govindarajan K, Leong B, Shahab A, Ruan Y, Bourque G, Sung W, Clarke N, Wei C, Ng H (2008) Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. *Cell* 133(6):1106–1117
20. Wang J, Rao S, Chu J, Shen X, Levasseur D, Theunissen T, Orkin S (2006) A protein interaction network for pluripotency of embryonic stem cells. *Nature* 444(7117):364–368
21. Niwa H, Miyazaki J, Smith AG (2000) Quantitative expression of Oct-3/4 defines differentiation, dedifferentiation or self-renewal of ES cells. *Nat Genet* 24(4):372–376
22. Kim J, Orkin SH (2011) Embryonic stem cell-specific signatures in cancer: insights into genomic regulatory networks and implications for medicine. *Genome Med* 3(11):75
23. Pearl J, Lee A, Leveson-Gower D, Sun N, Ghosh Z, Lan F, Ransohoff J, Negrin R, Davis M, Wu J (2011) Short-term immunosuppression promotes engraftment of embryonic and induced pluripotent stem cells. *Cell Stem Cell* 8(3):309–317

# Extremal Trees of the Reformulated and the Entire Zagreb Indices



Anjusha Asok<sup>ID</sup> and Joseph Varghese Kureethara<sup>ID</sup>

**Abstract** The first reformulated Zagreb index of trees can take any even positive integer greater than 8, whereas the second reformulated Zagreb index of trees can take all positive integers greater than 47 with a few exceptional values less than 8 and 47, respectively. The entire Zagreb index is defined in terms of edge degrees and incorporates the idea of intermolecular forces between atoms along with atoms and bonds. This intricate significance of studying the entire Zagreb index led to the generalization of the first entire Zagreb index of trees. The inverse problem on the first entire Zagreb of trees gives the existence of a tree for any even positive integer greater than 46.

**Keywords** Topological index · Molecular graph · Forgotten Zagreb index · Hyper Zagreb index first reformulated Zagreb index · Second reformulated Zagreb index · First entire Zagreb index

## 1 Introduction

Chemical graph theory is a branch of mathematics that assists in identifying and analyzing the properties of chemical compounds using specific molecular descriptors. Molecular descriptors are the output of a rational and analytical approach of mathematics to chemistry.

Topological indices are molecular descriptors enumerated from molecular graphs and analyzed to investigate the physio-chemical properties of chemical compounds. They are mainly categorized based on degree, distance, and spectrum of graphs. The most celebrated and widely discussed Wiener index is based on the distance of vertices in the respective graph and was defined by Wiener in 1947 [1]. It was found that since then, more than 3000 topological indices have been defined. Zagreb indices have gained special attention among the degree and distance-based topological indices among mathematicians and chemists.

---

A. Asok · J. V. Kureethara (✉)  
Christ University, Bengaluru, Karnataka, India  
e-mail: [fjoseph@christuniversity.in](mailto:fjoseph@christuniversity.in)

In this paper, we study the characteristics of two Zagreb indices: reformulated and entire Zagreb indices. The first objective of our article is to find a simple connected graph, given an integer value for the topological indices of the first reformulated Zagreb index, the second reformulated Zagreb index, and the first entire Zagreb index. The second objective is to characterize the first reformulated Zagreb index, the second reformulated Zagreb index, and the first entire Zagreb index for trees. These are achieved in the following sections using various transformations and some existing results. The first and second sections include the introduction and a few articles related to the study, respectively. Various graph operations and propositions on the topological indices of our interest are given in Sect. 3. In Sect. 4, some important properties and relations connecting the first reformulated Zagreb index, the second reformulated Zagreb index, and the first entire Zagreb index with few Zagreb indices and forgotten topological indices are given, which are used in the derivation of a few results of critical. Section 5 establishes the main results of the article. Finally, the last section concludes the study's significance and further research scope.

## 2 Reformulated and Entire Zagreb Indices

The Zagreb connection indices are topological indices [2] first seen in a paper published in the year 1972, in which the indices are used to compute the total electron energy of alternate hydrocarbons [3]. Forty years ago, the first and second Zagreb indices were recognized as group indices. Later, it was modified, redefined, and reformulated, resulting in many other topological indices. Though it did not influence the researchers initially, its applications in chemical graph theory flourished after a certain period of years. In 2004, the first and second reformulated Zagreb indices were introduced [4]. The first and second reformulated Zagreb indices are derived from the original Zagreb indices, replacing vertex degrees with edge degrees of the graph. In [5], properties of reformulated Zagreb index are investigated. Most articles like [6–9] compute the bounds of the reformulated Zagreb indices. Reformulated Zagreb index has been studied for compounds like dendrimers [10, 11] as well for different classes of graphs such as line graphs and edge-semi-total graphs [12]. It is found that inverse problems on topological indices are the least discussed among researchers. Let  $d(e)$  and  $d(f)$  be the degrees of edge e and f of the graph  $G$ , respectively. Then the first and the second reformulated Zagreb indices are defined, respectively as:

$$EM_1(G) = \sum_{e \in E(G)} (d(e))^2 \quad (1)$$

$$EM_2(G) = \sum_{e \sim f} d(e)d(f) \quad (2)$$

where  $e \sim f$  hints that  $e$  and  $f$  are adjacent edges in graph  $G$ . Further, we have  $d(e) = d(u) + d(v) - 2$  where  $d(u)$  and  $d(v)$  are the degrees of vertices  $u$  and  $v$  with  $e = uv$ .

It is found that the history of Zagreb indices until 2018 ripples around the construction of Zagreb indices based on the intermolecular forces between the atoms in a molecule. The disintegration of this traditional thought by Cangul et al. [13] in 2018 resulted in the first and second entire Zagreb indices. These indices inhibit the intermolecular forces between atoms and bonds and intermolecular forces among the atoms. It was recently entire Zagreb indices were defined. To know more about the different properties and bounds of entire Zagreb indices, refer to [13–15]. Let  $G$  be a graph with vertex set  $V(G)$  and edge set  $E(G)$ . Then the first and second entire Zagreb indices are defined, respectively, as follows:

$$M_1^e(G) = \sum_{x \in V(G) \cup E(G)} d(x)^2 \quad (3)$$

$$M_2^e(G) = \sum d(x)d(y) \quad (4)$$

where  $x$  and  $y$  are either adjacent or incident to each other.

### 3 Some Graph Operations

**Transformation 1:** Let  $G$  be a graph with  $n$  vertices and  $m$  edges, then we subdivide an edge  $uv$  of graph  $G$  by adjoining an edge  $e = xy$  resulting in a new graph  $G^*$ .

**Transformation 2:** Let  $G$  be a graph with  $n$  vertices and  $m$  edges, then we subdivide an edge  $uv$  of graph  $G$  by introducing a vertex  $w$  resulting in a new graph  $G^*$ .

**Proposition 1** *Given a graph  $G$  with an edge  $uv$  and first reformulated Zagreb index  $EM_1(G)$ ,  $EM_1(G)$  together with the Transformation 1 is given by,*

$$EM_1(G^*) = EM_1(G) + 6[d(u) + d(v)] - 2[d(u)d(v) - 1] \quad (5)$$

**Proof** Let  $G$  be the given graph with edge  $e = uv$ . Then by Transformation 1, the edge  $uv$  of graph  $G$  is replaced by a new set of edges say  $e_1 = ux$ ,  $e_2 = vx$  and  $e_3 = xy$  with  $d(x) = 3$  and  $d(y) = 1$ .

By the definition of the first reformulated Zagreb index and  $d(e) = d(u) + d(v) - 2$ , we have the following.

$$\begin{aligned}
EM_1(G^*) &= EM_1(G) + d(e_1)^2 + d(e_2)^2 + d(e_3)^2 - d(e)^2 \\
&= EM_1(G) + [d(u) + d(x) - 2]^2 + [d(v) + d(x) - 2]^2 \\
&\quad + [d(x) + d(y) - 2]^2 - [d(u) + d(v) - 2]^2 \\
&= EM_1(G) + [d(u) + 1]^2 + [d(v) + 1]^2 + [2]^2 - [d(u) + d(v) - 2]^2 \\
&= EM_1(G) + 6[d(u) + d(v)] - 2[d(u)d(v) - 1]
\end{aligned}$$

**Corollary 1** Let  $G$  be a graph with the first reformulated Zagreb index  $EM_1(G)$ . Then the new graph  $G^*$  obtained by Transformation 1 with  $d(u)$  or  $d(v)$  is equal to either 1 or 2 has

$$EM_1(G^*) = EM_1(G) + 8 \quad (6)$$

**Corollary 2** Let  $G$  be a graph with the first reformulated Zagreb index  $EM_1(G)$ . Then the new graph  $G^*$  obtained by Transformation 1 with  $d(u)=d(v)=2$  has

$$EM_1(G^*) = EM_1(G) + 18 \quad (7)$$

**Proposition 2** Given a graph  $G$  with an edge  $uv$  and first reformulated Zagreb index  $EM_1(G)$ ,  $EM_1(G^*)$  together with the Transformation 2 is given by,

$$EM_1(G^*) = EM_1(G) + 4[d(u) + d(v)] - 2[d(u)d(v) + 2] \quad (8)$$

**Proof** we prove this, similar to the above proposition. Let  $G$  be the given graph with edge  $e = uv$ . Then by the definition of the first reformulated Zagreb index and using  $d(e) = d(u) + d(v) - 2$ , we have

$$\begin{aligned}
EM_1(G^*) &= EM_1(G) + d(uw)^2 + d(vw)^2 + -d(uv)^2 \\
&= EM_1(G) + [d(u) + d(w) - 2]^2 + [d(v) + d(w) - 2]^2 \\
&\quad - [d(u) + d(v) - 2]^2 \\
&= EM_1(G) + [d(u) + 2]^2 + [d(v) + 2]^2 - [d(u) + d(v) - 2]^2 \\
&= EM_1(G) + 4[d(u) + d(v)] - 2[d(u)d(v) + 2]
\end{aligned}$$

**Corollary 3** Let  $G$  be a graph with  $n$  vertices. Choose an edge  $uv$  with  $d(u)$  and  $d(v)$  both equal to 2 or either of them is 1 or 2, then the newly generated graph  $G^*$  whose first reformulated Zagreb index is increased by 4 under the Transformation 2. That is,

$$EM_1(G^*) = EM_1(G) + 4 \quad (9)$$

**Proposition 3** Let  $G$  be a graph with first entire Zagreb index  $M_1^e(G)$  and  $G^*$  be the result of  $G$  under Transformation 2 with  $d(u)=d(v)=2$  or either  $d(u)$  or  $d(v)$  is equal to 1 or 2. Then

$$M_1^e(G^*) = M_1^e(G) + 8 \quad (10)$$

**Proof** From [15], we have

$$M_1^e(G) = M_1(G) + EM_1(G) \quad (11)$$

It is found that the value of the first Zagreb index  $M_1(G)$  in [16], is increased by 4 by means of Transformation 2. Therefore by Corollary 3 we have,

$$\begin{aligned} M_1^e(G^*) &= M_1(G^*) + EM_1(G^*) \\ &= M_1(G) + 4 + EM_1(G) + 4 \\ &= M_1^e(G) + 8 \end{aligned}$$

**Proposition 4** Let  $G$  be a graph with the second reformulated Zagreb index  $EM_2(G)$ . Choose a path  $P = uvwz$  with  $d(u) = 1$ ,  $d(v) = d(w) = 2$  in  $G$ . Now the subdivision of  $uv$  by a vertex, say  $x$ , will increase the second reformulated Zagreb index by 4, that is

$$EM_2(G^*) = EM_2(G) + 4 \quad (12)$$

where  $EM_2(G^*)$  is the new second reformulated Zagreb index.

**Proof** By definition,

$$\begin{aligned} EM_2(G^*) &= EM_2(G) + d(ux)d(xv) + d(xv)d(vw) - d(uv)d(vw) \\ &= EM_2(G) + [d(u) + d(x) - 2][d(x) + d(v) - 2] \\ &\quad + [d(x) + d(v) - 2][d(v) + d(w) - 2] \\ &\quad - [d(u) + d(v) - 2][d(v) + d(w) - 2] \\ &= EM_2(G) + (1)(2) + (2)(2) - (1)(2) \\ &= EM_2(G) + 4 \end{aligned}$$

**Proposition 5** Let  $uvwz$  be a path in  $G$  with  $d(u)=1$ ,  $d(v)=d(w)=2$ , then by the subdivision of the edge  $uv$  by a vertex, say  $x$  will result in a new graph  $G^*$  with second entire Zagreb index,

$$M_2^e(G^*) = M_2^e(G) + 16 \quad (13)$$

**Proof** From [15, 16] and Proposition 5 we have the following,

$$\begin{aligned} M_2^e(G^*) &= 3M_2(G^*) + EM_2(G^*) + F(G^*) - 2M_1(G^*) \\ &= 3[M_2(G) + 4] + [EM_2(G) + 4] + [F(G) + 8] - 2[M_1(G) + 4] \\ &= 3M_2(G) + EM_2(G) + F(G) - 2M_1(G) + 16 \\ &= M_2^e(G) + 16 \end{aligned}$$

## 4 Some Properties of Zagreb Indices

We present some of the feasibilities of a selected number of topological indices in the next table.

Summary of the properties				
Name of the index	Notation	Formula	Co-domain	References
The first Zagreb index	$M_1(G)$	$\sum_{u \in V} d_u^2$ [17]	$2\mathbb{Z}^+$	[16]
Forgotten topological index	$F(G)$	$\sum_{u \in V} d_u^3$ [17]	$2\mathbb{Z}^+$	[16]
The second Zagreb index	$M_2(G)$	$\sum_{uv \in E} d_u d_v$ [16]	$\mathbb{Z}^+$	[16]
The first reformulated Zagreb index	$EM_1(G)$	$\sum_{e \in E(G)} d_e^2$ [4]	$2\mathbb{Z}^+$	New
The second reformulated Zagreb index	$EM_2(G)$	$\sum_{e \sim f} d_e d_f$ [4]	$\mathbb{Z}^+$	New
The first entire Zagreb index	$M_1^e(G)$	$\sum_{x \in V \cup E} d_x^2$ [13]	$2\mathbb{Z}^+$	New
The second entire Zagreb index	$M_2^e(G)$	$\sum_{e \sim f} d_e d_f$ [13]	$\mathbb{Z}^+$	New

For the connected graphs, the first Zagreb index,  $M_1(G)$ , the forgotten topological index,  $F(G)$ , the first reformulated Zagreb index,  $EM_1(G)$ , and the first entire Zagreb index,  $M_1^e(G)$ , can take only even positive integers. However, the second Zagreb index,  $M_2(G)$ , the second reformulated Zagreb index,  $EM_2(G)$  and the second entire Zagreb index  $M_2^e(G)$ , can take any positive integer.

## 5 Main Results

**Theorem 1** *The first reformulated Zagreb index  $EM_1$  of trees can take all even integer values of  $x$  such that  $x \notin \{4, 8\}$  and  $x \geq 0$ .*

**Proof** Consider two non-empty sets say,

$$S_1 = \{x : x \in [0]_4, \quad x \geq 12\} \quad (14)$$

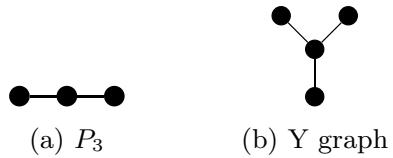
and

$$S_2 = \{x : x \in [2]_4, \quad x \geq 2\} \quad (15)$$

where  $[0]_4$ ,  $[2]_4$  are residue class modulo 4. To prove this theorem, we show that the first reformulated Zagreb index  $EM_1$  of trees can take all the values in the set  $S_1$  and  $S_2$ . Further, we show trees cannot take 4 and 8.

Let  $G$  be a tree with the first reformulated Zagreb index  $EM_1$  on  $n$  vertices and  $m$  edges and  $EM_1^*$  be the first reformulated Zagreb index of  $G$  under the Transformation

**Fig. 1** Two fundamental trees for the first reformulated Zagreb index



**Table 1** Edge degree sequence of possible graph with  $EM_1 = 4$  and 8

$EM_1$	Degree sequence of edges	
4	(1,1,1,1)	(2)
8	(1,1,1,1,1,1,1,1)	(2, 1, 1, 1, 1) (2,2)

2. Consider the following trees as shown in Fig. 1a, b on  $n = 3$  (path  $P_3$ ) and  $n = 4$  (Y graph) vertices with  $EM_1$  equal to 2 and 12, respectively.

Let us start with an edge  $e = uv$  in  $P_3$  whose end vertices have degrees equal to 2 or either 1 or 2. Construct a new graph  $G^*$  from  $P_3$  by the subdivision of edge  $e$  in  $G$  by Transformation 2. Then the first reformulated Zagreb index of the new graph  $G^*$  is increased by 4 using Corollary 3 and therefore  $EM_1(G^*) = 6$  with  $n = 3$ . The repeated subdivision of edge  $e$  in  $P_3$  of our concern constructively builds graphs with the first reformulated Zagreb index 2, 6, 10, 14 and so on. This way we can achieve at least one graph with the first reformulated Zagreb index whose value belongs to the set  $S_1$ . Similarly, we chose an edge in the Y-graph with the same requirements as mentioned in Corollary 3 to go under Transformation 2. Being  $EM_1(Y - \text{graph}) = 12$ , the Y-graph can act as a generator for all graphs with the first reformulated Zagreb index equal to 16, 20, 24 and thereby all values in the set  $S_2$ .

Now we need to show that there are no graphs with  $EM_1$  equal to 4 and 8. By the definition of the first reformulated Zagreb index, we have the following possible edge degree sequences for graphs with  $EM_1 = 4$  and 8 as mentioned in Table 1. Therefore, we show that there are no trees with edge degree sequence as mentioned in Table 1.

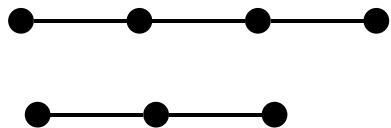
We know every tree on  $n$  vertices contains exactly  $n-1$  number of edges. This fact about trees proves that there are no trees with edge degree sequence {2} and {2, 2} as the only tree on 2 vertices is a path  $P_2$  with an edge of degree 0 and on 3 vertices is a path  $P_3$  with edge degree sequence {1, 1}. A graph with all edges of degree 1 is a disconnected graph whose components are  $P_3$  for all  $n \geq 4$ . Also, the only possible graph with edge degree sequence {2, 1, 1, 1, 1} is again a disjoint union of  $P_2$  and  $P_3$  whose components are paths on  $n = 3$  and 4 as shown in Fig. 2. This shows that there are no trees with the first reformulated Zagreb index equal to 4 and 8.

Hence, the theorem is proved.

**Theorem 2** *The first entire Zagreb index of trees can take all even positive integers except 4, 6, 10, 12, 14, 18, 22, 26, 28, 30, 36, 38, and 46.*

**Proof** We know that the first entire Zagreb index of trees can take only even positive integers. So it is enough to show that the first entire Zagreb index of trees can take

**Fig. 2** Graph with edge degree sequence  
 $\{2, 1, 1, 1, 1\}$



**Table 2** Integer values for which the first entire Zagreb index of tree does not exist

$E_1$	$E_2$	$E_3$	$E_4$
		4	6
	10	12	14
	18	20	22
	26	28	30
		36	38
			46

integers greater than 46 and cannot take integers, as mentioned in Table 2. We prove this theorem in two parts: the first part of the theorem shows that the first entire Zagreb trees can take only integers greater than 46. Later in the second part of the proof, we establish the non-existence of trees with the first entire Zagreb index, as mentioned in Table 1.

For this, we divide the entire set of positive integers into four different sets,  $E_1$ ,  $E_2$ ,  $E_3$  and  $E_4$  as follows where  $[0]_8$ ,  $[2]_8$ ,  $[4]_8$  and  $[6]_8$  are residue class modulo 8.

$$E_1 = \{x : x \in [0]_8, x \geq 0\} \quad (16)$$

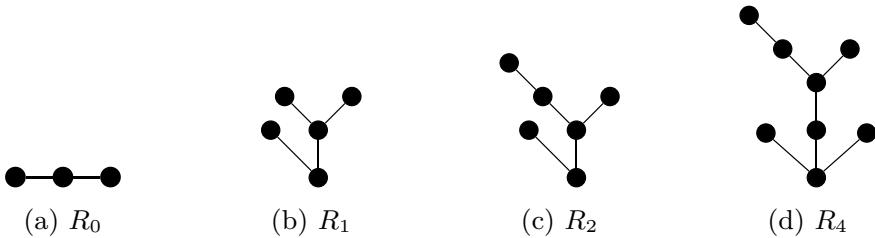
$$E_2 = \{x : x \in [2]_8, x \geq 0\} \quad (17)$$

$$E_3 = \{x : x \in [4]_8, x \geq 0\} \quad (18)$$

$$E_4 = \{x : x \in [6]_8, x \geq 0\} \quad (19)$$

As shown in Table 2, all values in the set  $E_1$  are taken by the first entire Zagreb index for trees whereas integers less than 26, 36 and 46 in the set  $E_2$ ,  $E_3$ ,  $E_4$ , respectively, are not taken. So let us start with set  $E_1$ . For this, consider path  $R_0$  with  $M_1(R_0) = 6$  by definition and choose an edge with a leaf in  $R_0$ . Add a vertex into the edge, this will increase the value of the first entire Zagreb index of  $R_0$  by 8 using Proposition 3. This process of repeated addition of vertex into the edge produces paths with the first entire Zagreb index 16, 24, 32, and so on and therefore trees with the first entire Zagreb index in  $E_1$ .

Now let us consider the graphs in Fig. 3a–d and follow similar steps used in the construction of trees with the first entire Zagreb index in set  $E_1$ , such as choosing an edge in the graph as required to apply Proposition 3, a subdivision of the edge and repeated subdivision of the edge by introducing a vertex at each step of subdivision.



**Fig. 3** Four fundamental trees for the first entire Zagreb index

Similarly, we consider the subdivision graphs of  $R_1, R_2, R_3$ . The repeated subdivision of an edge in  $R_1, R_2, R_3$  helps to obtain graphs with the first entire Zagreb index greater than 26, 36 and 46  $E_2, E_3$  and  $E_4$ , respectively.

The second part of the theorem is proved in different cases as follows:

Case 1:  $M_1^e(G) \neq 4, 6$

Consider the set of trees with  $p \leq 3$ . The only trees with order 1, 2 and 3 are all path graphs with  $M_1^e(G) = 0, 2$  and 8, respectively. By the definition of the first entire Zagreb index, the numerical value of  $M_1^e(G)$  increases with an increase in the number of vertices for trees. Therefore, it is trivial that no trees with  $M_1^e(G) = 4$  and 6 exist.

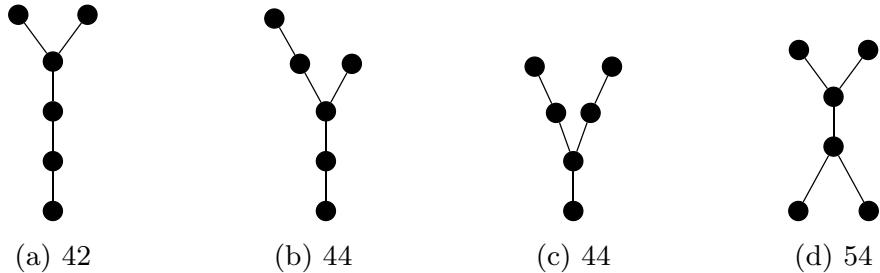
Case 2:  $M_1^e(G) \notin \{10, 12, 14, 18, 22, 26, 28, 30\}$

To prove case 2, we start with  $P_6$ . The first entire Zagreb index for  $P_6$  is 32. The smallest value for the first entire Zagreb index for any tree of order  $n$  is for a path. That is no trees of order  $n \geq 6$  can take values greater than 32. So we have to show that no trees of order  $n = 4$  or 5 can take integers mentioned in the set. For  $n = 4$  there are only two trees possible, a path  $P_4$  with  $M_1^e(G) = 16$  and a star graph  $S_4$  with  $M_1^e(G) = 24$ . We know that for any  $n$ , the only tree with  $\Delta = 2$  and  $\Delta = n - 1$  is a path and star graph. Let  $n = 5$ , then  $M_1^e(P_5) = 24$  ( $\Delta = 2$ ) and  $M_1^e(S_5) = 56$  ( $\Delta = 4$ ). Now using the table of properties, the first entire Zagreb index of trees with  $\Delta = 3$  can take only values greater than or equal to 34. This proves that no trees with  $M_1^e(S_5) \leq 32$  of order 5 thereby we proved case 2.

Case 3:  $M_1^e(G) \notin \{36, 38, 46\}$

The first entire Zagreb index of path  $P_8$  is equal to 48. Path being the graph with the smallest integer value for the first entire Zagreb index for any order  $n$ , no trees of order  $n \geq 8$  contain graphs with the first entire Zagreb index equal to 36, 38, and 46. Therefore, we now look into graphs of order  $n = 6$  and  $n = 7$ . We start with the integer values 36 and 38. Since the least integer value taken by trees of order  $n = 7$  is 46, we can now check for trees of order  $n = 6$ . When  $n = 6$  with  $\Delta = 3$ , it is a path  $P_6$  with  $M_1^e(P_6) = 32$ . Now using the table of properties, we can say trees of order  $n = 6$  with  $\Delta = 3$  has  $M_1^e(G) \geq 42$ . This implies that no trees with  $n = 6$  take first entire Zagreb index as 36 or 38.

Now let us look into graphs of order  $n = 7$  with  $\Delta = 3$ . The least possible value for the first entire Zagreb index is 50. Therefore, no tree exists with  $M_1^e(G) = 46$ . Consider trees of order 6 and  $\Delta = 3, 4$  and 5. Using the table of properties, a tree of



**Fig. 4** Trees of order 6 with  $\Delta = 3$  and its first entire Zagreb index

order 6 with  $\Delta = 4$  has  $M_1^e(G) \geq 64$  and the only tree with  $\Delta = 5$  is a star graph with  $M_1^e(G) = 40$ . This leaves us behind trees of order  $n = 6$  and  $\Delta = 3$ . The first entire Zagreb index of trees of order  $n = 6$  and  $\Delta = 3$  are mentioned in the table below, which helps to conclude case 3 shown in Fig. 4a-d.

Part 1 of the proof and three cases establish the theorem.

**Theorem 3** If  $G = S_n$  or  $P_n$ , then the second reformulated Zagreb index of star graphs are given by  $EM_2(G) = \frac{(n-2)^3(n-1)}{2}$  and path graphs are given by  $EM_2(G) = 4(n-3)$ .

**Proof** Let  $G = S_n$  be a star graph with  $n$  number of vertices and  $m$  number of edges. In a star graph, every edge is adjacent to every other edge. Therefore, edge degree of an edge is  $m-1$ . By the definition of the second reformulated Zagreb index, every term is the product of the edge degree of pair of edges adjacent to each other, which is  $(m-1)^2$  for a star graph. The number of ways the edges are adjacent provides you with the total number of pairwise adjacent edges, given by  ${}^m C_2$ . This follows:

$$\begin{aligned} EM_2(G) &= {}^m C_2(m-1)^2 \\ &= \frac{m(m-1)}{2}(m-1)^2 \\ &= \frac{m(m-1)^3}{2} \\ &= \frac{(n-1)(n-2)^3}{2} \end{aligned}$$

For any path  $G = p_n$ , there is exactly two leaf which is of degree 1 and the remaining  $m-3$  edges have a degree exactly equal to 2. Then by the definition of the second reformulated Zagreb index, we have

**Table 3** Integer values for which the second reformulated Zagreb index of tree does not exist

$S_1$	$S_2$	$S_3$	$S_4$
	2	3	
5	6	7	
9	10	11	
13	14	15	
17	18		
21	22	23	
25	26		
29	30	31	
33	34	35	
	38	39	
	42	43	
		47	

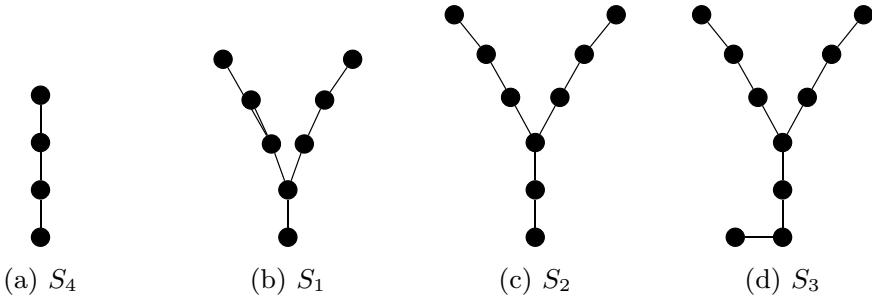
$$\begin{aligned} EM_2(G) &= 4(m - 3) + 2(2) \\ &= 4(n - 3) \end{aligned}$$

**Theorem 4** *The second reformulated Zagreb indices of trees can take any positive integer except 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15, 17, 18, 21, 22, 23, 25, 26, 29, 30, 31, 33, 34, 35, 38, 39, 42, 43, 47.*

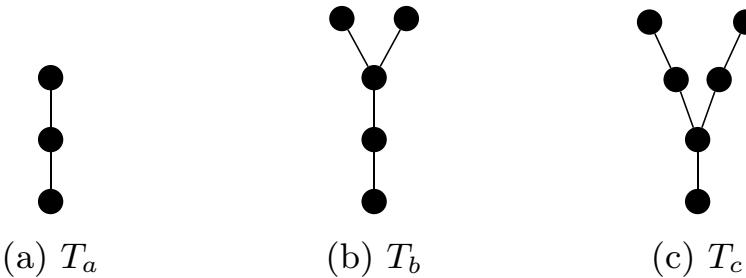
**Proof** In order to prove this theorem, we need to show that there are no trees with the second reformulated Zagreb index equal to 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15, 17, 18, 21, 22, 23, 25, 26, 29, 30, 31, 33, 34, 35, 38, 39, 42, 43, 47. Also show that for a given positive integer greater than 47 and some few integers less than 47, there exists a tree with the second reformulated Zagreb index equal to integers not mentioned in Table 3.

It is clear that the second reformulated Zagreb index can take any even positive integer. Now we show that it can take any positive integer greater than 47. This proof is similar to the previous theorem. All the graphs that have been taken to generate trees with the second reformulated Zagreb index greater than 47 contain a path  $P_4$  resilient to constraints as mentioned in Proposition 4.

Consider the graphs  $S_1, S_2, S_3$  and  $S_4$  in the Fig. 5a–d. Let  $G=S_4$  with  $EM_2(G)=4$ . Choose a path  $P = uvwz$  with  $d(u) = 1, d(v) = d(w) = 2$  in  $G$ . Now subdivide the edge  $uv$  in  $G$  by introducing a new vertex into the graph. This increases the second reformulated Zagreb index of  $G$  by 4 and thereby generates a new graph  $G^*$  with  $EM_2(G^*) = 8$  by Proposition 4. The repeated subdivision of the edge  $uv$  will produce graphs with the second reformulated Zagreb index 4, 8, 12, and so on. That is, given an integer of form  $4p + 4$  where  $p \geq 0$  and  $p \in \mathbb{Z}^+$ , we can always find a tree with the second reformulated Zagreb index equal to the integer value using the



**Fig. 5** Parent graphs for  $EM_2(G) \geq 51$



**Fig. 6** Special graphs with second reformulated Zagreb index 1, 19, and 27

parent graph  $S_4$ . Similarly, if we consider the parent graphs  $S_1$ ,  $S_2$  and  $S_3$  with second reformulated Zagreb index 8, 46 and 51, respectively, we can construct graphs with second reformulated Zagreb index for any positive integer of form  $37p + 4$ ,  $46p + 4$  and  $51p + 4$  where  $p$  is the number of subdivisions, i.e.,  $S_4$  can generate trees with second reformulated Zagreb index equal to 4, 8, 12, 16,...,  $S_1$  can generate trees with second reformulated Zagreb index equal to 37, 41, 45, 49,...,  $S_2$  and  $S_3$  can generate trees with second reformulated Zagreb index equal to 46, 50, 54, 58,...and 51, 55, 59, 61,..., respectively. This way, we can show that the second reformulated Zagreb index can take any positive integer greater than 47. Also, we have a few integers for which trees with the second reformulated Zagreb index of less than 47 exist. Figure 6a–c gives special trees with the second reformulated Zagreb index equal to 1, 19, and 27, respectively, which are not covered by the parent graphs.

Consider the sets A, B, C and D where  $A = \{2, 3, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15\}$ ,  $B = \{17, 18, 21, 22, 23, 25, 26\}$ ,  $C = \{29, 30, 31, 33, 34, 35\}$  and  $D = \{39, 42, 43, 47\}$ . The next part of the theorem shows that trees cannot take integers from sets A, B, C, and D.

Let us begin with set A. By the definition of the second reformulated Zagreb index, the path has the smallest second reformulated Zagreb index of all trees for a given  $n$ . For  $n = 7$ ,  $EM_2(P_7) = 16$ . This implies that no trees of order  $n \geq 7$  can take the second reformulated Zagreb index less than 16 and, therefore, no integer

**Table 4** Second reformulated Zagreb index of parent graphs and special graphs

$S_1$	$S_2$	$S_3$	$S_4$	$T_a$	$T_b$	$T_c$
37	46	51	4	1	19	27

**Table 5** Second reformulated Zagreb index for trees of order  $n = 4$  to  $n = 7$ 

order	4	5	6	7
	4	8	12	16
	12	19	24	28
		54	27	32
			27	36
			40	73
			67	81
			160	165
				375

values from set A. The second reformulated Zagreb index is a measure of adjacency between edges; it is valid only for trees of order  $n \geq 3$ . The only tree with  $n = 3$  is a path graph with  $EM_2(P_3) = 1$ . The second reformulated Zagreb index of all trees of order  $n=4$  to  $n=6$  is given in Table 4. It is evident that no trees of order  $n = 3$  to  $n = 6$  take any integer values for the second reformulated Zagreb index from set A.

Consider set B. For  $n = 10$ ,  $EM_2(P_{10}) = 28$ . Again path being the tree with the smallest second reformulated Zagreb index for a given  $n$ , no trees of order  $n \geq 10$  exist with  $EM_2 \leq 28$ . Table 5 shows no trees of order  $n = 3$  to  $n = 6$  take values from set B. Therefore, it is enough to show that trees of order 7, 8, and 9 do not contain trees with the second reformulated Zagreb index from set B. For a given  $n$ , the next smallest integer value for the second reformulated Zagreb index is given by graphs of the form  $G = P_{n-4} + k_{1,3}$ . This graph  $G$  is obtained by applying Transformation 1 in a path graph  $P_{n-1}$  at the vertex adjacent to the leaf. The resultant graph  $P_{n-4} + k_{1,3}$  has the second reformulated Zagreb index of form  $4n$  for a given  $n$ . This formula is only applicable for  $n \geq 5$ . We have computed the second reformulated Zagreb index for trees of order  $n = 7$  to  $n = 12$  of the form  $P_{n-4} + k_{1,3}$  and is given in Table 6. The maximum value in set B is 26, whereas the second smallest integer value for the second reformulated Zagreb index is 28, 32, and 40 for  $n = 7, 8$ , and 9, respectively. This proves that no tree of order  $n = 3$  to  $n = 9$  can take integers from set B.

For a given  $n$ , we obtain a tree with the third smallest integer for the second reformulated Zagreb index by adjoining an edge to a vertex in  $P_{n-1}$  adjacent to the pendant edge. Let us call this graph  $T_d$ . The second reformulated Zagreb index for  $T_d$  is given by  $4(n+1)$  for  $n \geq 8$ . Again no trees of order  $n \geq 12$  contain trees with the second reformulated Zagreb index from set C as  $EM_2(P_{12}) = 36$ . This reduces the order from  $n = 3$  to  $n = 11$ . Tables 5 and 6 show no tree exists with the second reformulated Zagreb index from set C.

**Table 6** Second reformulated Zagreb index for  $P_n$  and  $P_{n-4} + k_{1,3}$  of order  $n = 7$  to  $n = 12$ 

n	7	8	9	10	11	12
$P_n$	16	20	24	28	32	36
$P_{n-4} + k_{1,3}$	28	32	36	40	44	48
$T_d$		36	40	44	48	52

Finally, the last set,  $D$ . In order to show that there exist no trees with the second reformulated Zagreb index from set  $D$ , we divide this set into three subsets, say  $d_1 = \{39\}$ ,  $d_2 = \{42, 43\}$  and  $d_3 = \{47\}$ . Since  $EM_2(P_{15}) = 48$ , we can ensure that trees of order greater than or equal to 15 cannot take positive integers for the second reformulated Zagreb index from set  $D$ . Also, Table 5 shows that the second reformulated Zagreb index of no trees of order  $n = 3$  to 7 belongs to set  $D$ . This leaves us behind with trees of order  $n = 8$  to  $n = 14$ .

Consider the subset  $d_1 = \{39\}$ ,  $d_2 = \{42, 43\}$  and  $d_3 = \{47\}$ . From Table 6, it is very clear that the first and second smallest integer values of the second reformulated Zagreb index do not belong to  $d_1$ ,  $d_2 = \{42, 43\}$  and  $d_3 = \{47\}$ . Further, the third smallest integer values for all orders greater than or equal to 11 are greater than 47 (see Table 6). This brings the  $n$  value to 8 to 10 from 8 to 14. In addition to the above arguments, for a given  $n$ , the only tree with  $\Delta = 2$  is a path  $P_n$  and the second reformulated Zagreb index for all trees with  $\Delta \geq 4$  should be greater than 63. This shows that we need to check only order 8, 9, and 10 graphs with  $\Delta = 3$ . This is because as  $\Delta$  increases, the second reformulated Zagreb index of trees also increases by its definition. The second reformulated Zagreb index of trees with  $n = 8$  and  $\Delta = 3$  except  $S_1$  are 32, 36 and 41. Similarly, the second reformulated Zagreb index for trees with  $n=9$  and 10 with  $\Delta = 3$  except  $S_2$  and  $S_3$  are 36, 40, 41, 45 and 40, 44, 45, 49, 45, respectively. This completes the proof.

## 6 Conclusion

This study the inverse problems for the topological indices, such as the first and the second reformulated Zagreb indexes and the first entire Zagreb index. We also present results on graph transformations of trees connecting the topological indices of our concern with various other Zagreb indices. A significant point of chemical graph theory is to concoct valuable graph invariants, which have predictive power for the physio-chemical properties of the molecule. In chemical graph theory, such graph invariant is called a topological index. The QSPR studies and the study of topological indices for different classes of graphs (molecular) have become powerful tools in contemporary chemical and medicinal research, as they have made it possible to predict the biological activity, specific chemical activity, toxicity, etc., for the chemical compounds without conducting real-time experiments. This method of

identifying the physio-chemical properties of chemical compounds is cost-effective and less time-consuming. Therefore, the need to study different topological indices and their characterization is also considered vital, specifically Zagreb indices, which correlate well with many physio-chemical properties of chemical compounds. These results can be further used in the inverse problem on the second entire Zagreb indices of trees and any other class of simple connected graphs.

## References

1. Harry W (1947) Structural determination of paraffin boiling points. *J Am Chem Soc* 69(1):17–20
2. Sonja N, Goran K, Ante M, Nenad T (2003) The Zagreb indices 30 years after. *Croatica Chem Acta* 76(2):113–124
3. Cao J, Ali U, Javaid M, Huang C (2020) Zagreb connection indices of molecular graphs based on operations. *Complexity*
4. Ante M, Sonja N, Nenad T (2004) On reformulated Zagreb indices. *Molecular Diversity* 8(4):393–399
5. Bo Z, Nenad T (2010) Some properties of the reformulated Zagreb indices. *J Math Chem* 48(3):714–719
6. Nilanjan D (2012) Some bounds of reformulated Zagreb indices. *Appl Math Sci* 6(101):5005–5012
7. Shengjin J, Xia L, Bofeng H (2014) On reformulated Zagreb indices with respect to acyclic, unicyclic and bicyclic graphs. *MATCH Commun Math Comput Chem* 72(3):723–732
8. Milovanović EI, Milovanović IŽ, Doličanin EĆ, Glogić E (2016) A note on the first reformulated zagreb index. *Appl Math Comput* 273:16–20
9. Qu T, Mengya H, Shengjin J, Xia L (2020) Note on the reformulated Zagreb indices of two classes of graphs. *J Chem* 1–4:2020
10. Nilanjan D (2013) Reformulated Zagreb indices of dendrimers. *Math Aeterna* 3(2):133–138
11. Husin MN, Hasni R, Imran M (2017) More results on computation of topological indices of certain networks. *Int J Network Virt Org* 17(1):46–63
12. Liu J-B, Ali B, Malik MA, Afzal Siddiqui HM, Imran M (2019) Reformulated Zagreb indices of some derived graphs. *Mathematics* 7(4):366
13. Alwardi A, Alqesmah A, Rangarajan R, Cangul IN (2018) Entire Zagreb indices of graphs. *Discrete Math Alg Appl* 10(03):1850037
14. Luo L, Dehgardi N, Fahad A (2020) Lower bounds on the entire zagreb indices of trees. *Discrete Dynam Nat Soc*
15. Ghalavand A, Reza Ashrafi A (2019) Bounds on the entire Zagreb indices of graphs. *MATCH Commun Math Comput Chem* 81:371–381
16. Yurtas A, Togan M, Lokesh V, Cangul IN, Gutman I (2019) Inverse problem for Zagreb indices. *J Math Chem* 57(2):609–615
17. Gutman I, Trinajstić N (1972) Graph theory and molecular orbitals. Total  $\varphi$ -electron energy of alternant hydrocarbons. *Chemical Phys Lett* 17(4):535–538

# Sign Language Interpreter Using Stacked LSTM-GRU



M. Dhilsath Fathima , R. Hariharan , Sachi Shome,  
Manbha Kharsyiemlieh, J. Deepa, and K. Jayanthi

**Abstract** Sign language has been used to communicate with people who are hard of hearing in conveying their thoughts and ideas to ordinary people. People may readily express thoughts using this sort of gesture-based language, which reduces barriers caused by hearing problems. The major issue is that the vast majority of the population lacks the knowledge of using sign language. Sign language detection from live video footage is a challenging problem that can bridge the communication gap. This paper proposes a method for identifying sign language motions in real-time video data that combines computer vision and deep learning approaches. The major contribution of the proposed model is stacking the long short-term memory (LSTM) and gated recurrent unit (GRU) architecture called LSTM-GRU to detect and classify signs from sign language videos. Our built model is run on a real-time video stream using the camera input, and by stacking the LSTM and GRU, we optimize model performance and reach 94.4% prediction accuracy.

**Keywords** Long short-term memory · Gated recurrent unit · Sign language · Mediapipe · OpenCV

---

M. Dhilsath Fathima ()

Department of Computational Intelligence, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India  
e-mail: [dilsathveltech123@gmail.com](mailto:dilsathveltech123@gmail.com)

R. Hariharan

Research Scholar, National Institute of Technology, Trichy, India

S. Shome · M. Kharsyiemlieh · J. Deepa

Department of Information Technology, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

K. Jayanthi

Department of Computer Science and Engineering, SRM Institute of Science and Technology, Tiruchirappalli, Tamil Nadu, India

## 1 Introduction and State of the Art in the Detection of the Sign Language

In sign language, body language, hand movements, and facial expressions are used to visually convey meaning. Learning sign language may be very helpful for people who have difficulty hearing or speaking. The process of turning these gestures into legally accepted spoken language words or alphabets is known as sign language recognition [1]. Sign language comprises a unique set of norms and syntax for effective communication. One of the most efficient methods for humans to communicate with others is through hand gestures that can be static or dynamic. Static gestures involve hand shape and dynamic gestures involve hand movements. Therefore, an algorithm or model that translates sign language into words helps close the communication gap between the hearing-impaired population. Sensor-based and image-based approaches are the two most used ways for recognizing sign language. However, research on image-based models is being performed purely for the benefit of not needing to use complex gear like Helmet, hand gloves and so on, as in sensor-based approaches.

Image comprehension is frequently linked to sign recognition. Sign recognition and sign detection are two processes. The process of extracting a feature from a particular object in respect to specific parameters is known as sign detection. The process of identifying a certain shape that sets an object apart from other shapes is referred to as sign recognition. The social orientation and meaning of sign language, as well as its practicality and use in terms of technology, are what set it apart from other forms of communication. There are a number of deep learning approaches that can be used for sign language detection. CNN is a kind of neural network primarily for image recognition. They function by learning to recognize patterns in visual representations, such as the contours of hands and fingers. Recurrent neural networks (RNNs) provide an effective approach for deep learning in sign language detection. Specifically designed to process sequential input, they excel in tasks that involve continuous monitoring of hand movements over time, making them well-suited for sign language interpretation. The difficulty with RNNs lies in managing long-term dependencies since they store information from earlier time steps in a single hidden state, leading to a decrease in the accuracy of knowledge within the hidden state over time. This limitation poses challenges in capturing long-range dependencies effectively. K-nearest neighbor (KNN) is a machine learning algorithm that can be used for sign language detection. KNN works by finding the  $k$  most similar training examples to a new test example, and then predicting the class of the test example based on the classes of the  $k$ -nearest neighbors.

Our proposed model uses a webcam to capture photos, Mediapipe and OpenCV preprocess the signs, and an LSTM-GRU interprets the gestures. A novel deep learning-based approach for sequential data is called LSTM-GRU classification. This makes it suitable for the natural language processing (NLP), where the arrangement of the words is important. RNNs are also a type of deep learning-based method that can be used for sequential data.

The proposed model aim is to empower individuals with hearing loss through a robust sign language detection system. Utilizing machine learning and Mediapipe, the goal is to recognize sign motions in real-time, fostering inclusivity in society. The absence of reliable sign language detecting technologies is a key obstacle that needs overcoming. The objective is to create a user-friendly and accurate system capable of interpreting a variety of sign language movements using advanced LSTM-GRU neural network techniques. By improving the lives of people with hearing impairments and promoting understanding, this project envisions a more accepting society for everyone.

Hearing loss is the most frequent sensory deficit in today's population. In India, the prevalence of Severe Auditory Impairment is estimated to be 6.3% of the population and affects approximately 63 million people. According to an NSSO study (NSSO, 2001), there are 291 individuals with moderate to profound hearing loss for every 100,000 people [2]. A significant portion of them are children, ranging in age from 0 to 14. A major loss in physical and economic output results from the large number of young Indians who are hearing-impaired.

The main problem is that people with hearing impairments find it difficult to engage with non-impaired people since they are not proficient in sign language. The solution is to develop an interpreter that can distinguish sign language used by disabled people, then input the sign into a machine learning algorithm, which is then acknowledged by the neural network and translated on the screen so that a non-disabled person can comprehend exactly what the sign means. Sign language is an independent natural language with its own structure and grammar, utilizing arm movements, hands, body, head movements, and facial expressions to convey semantic information and emotions.

The state of the art in detection of sign language is described below:

Tamiru et al. [3] developed a system for recognizing Amharic sign language utilizing ANN and SVM. This paper does not extensively address the possible challenges associated with implementing the system in real-world scenarios, including handling fluctuations in lighting conditions, camera angles, and background clutter. Akash et al. [4] developed real-time Bangla sign language (BSL) detection and sentence formation, utilizing action recognition techniques for gesture identification and sentence construction. A potential drawback of the paper is the lack of detailed discussion or evaluation of the model's performance under various real-world conditions.

Sreemathy et al. [5] proposed the utilization of artificial intelligence for sign language recognition. The study likely explores methods for automating sign language interpretation, beneficial for individuals with hearing impairments. A potential drawback is the limited exploration of challenges related to real-world implementation.

Aldhahri et al. [6] explore the utilization of convolutional neural networks and MobileNet for recognizing Arabic sign language. This approach has the potential to automate sign language interpretation, improving communication for individuals with hearing impairments. Incorporating deep learning methods provides the benefit of improved precision and efficiency in the recognition process.

Shin et al. [7] developed transformer-based deep neural networks for recognizing Korean sign language. This approach has the advantage of potentially capturing complex linguistic structures in sign language. A drawback arises from the substantial computational resources it demands. This may limit its practicality when implemented on specific devices or platforms.

Nandhi et al. [8] focus on recognizing the Indian sign language alphabet through convolutional neural networks (CNNs) with diffGrad optimizer and stochastic pooling. This innovative approach holds the potential to enhance recognition accuracy. However, the paper has limitations in terms of not extensively discussing potential drawbacks or challenges associated with their method, which could impact its real-world applicability and performance under various conditions.

### ***1.1 Benefits of LSTM-GRU Classification Over Other Deep Learning-Based Techniques***

- Better at managing long-term dependencies: LSTMs and GRUs were created with the intention of managing long-term dependencies, making them suitable for NLP tasks.
- More precise: For tasks involving sequential data, LSTMs and GRUs are typically more precise than other deep learning-based approaches.
- Robust to noise: LSTMs and GRUs are ideal options for situations where the data may be noisy since they are relatively resilient to data noise.

## **2 The Proposed Sign Language Interpreter**

The proposed method for a sign language interpreter involves using Mediapipe, an open-source library for extracting key point information from sign language videos and an LSTM-GRU model for training and prediction. Mediapipe provides pretrained models that can extract both hand and position landmarks at the same time. Hand landmarks are used to capture hand gestures, whereas pose landmarks are used to catch the signer's body position and movements, which are also significant for sign language detection.

The Hand and Pose Landmark models from Mediapipe use deep neural network architectures to recognize and localize hand and body areas in each frame, and then estimate keypoint positions in 3D space. The Hand Landmark model forecasts 21 key points for each hand, whereas the Pose Landmark model forecasts 33 key points for the entire body [9].

The LSTM-GRU model is the proposed neural network architecture that excels at sequence-to-sequence prediction tasks like converting sign language movements into spoken or written language [10]. The model is made up of LSTM and GRU layers that enable it to learn and remember long-term associations in sequence data.

The model predicts the spoken or written language translation using the Mediapipe keypoint data as input.

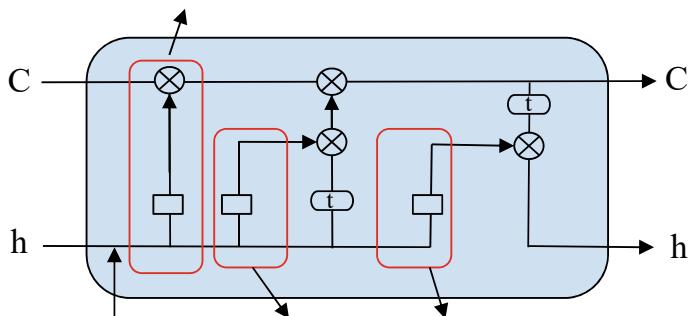
## 2.1 Preprocessing

Preprocessing plays an important role in the accurate and efficient detection of sign language gestures. In this study, we track hands and gestures in real-time and then extract key points using the Mediapipe framework. The first step of preprocessing involves feeding video frames captured by a camera or a recorded video source to the Mediapipe hand detection model. This model identifies and localizes the hand regions in each frame, enabling precise tracking of hand movements. The Mediapipe pose detection model is then used to calculate the 2D coordinates of important hand landmarks, including the joints in the fingers, knuckles, and wrist. These key points provide valuable spatial information necessary for accurately capturing the unique gestures of sign language.

## 2.2 Stacked LSTM-GRU Classifier

The LSTM is an RNN variant intended to address the vanishing gradient problem of typical RNNs [11]. An input gate, a forget gate, and an output gate comprise the LSTM. These gates enable the model to keep or forget information based on its relevance to the given task, which is very helpful in predicting long sequences. The graphical representation of the LSTM is shown in Fig. 1. The mathematical formulas for an LSTM network are as follows:

1. Input gate: The information entering the memory cell is managed by this gate. It receives as inputs the current input vector  $x_t$  and the prior hidden state  $h_{t-1}$ , and it produces the vector shown in Eq. (1), which specifies how much of the



**Fig. 1** LSTM

input should be allowed to pass.

$$it = \sigma(Wi[ht - 1, xt] + bi) \quad (1)$$

where  $Wi$ ,  $Ui$ , and  $bi$  are learnable weights and biases, and  $\sigma$  is the sigmoid function.

2. Forget gate: This gate regulates the data flow leaving the memory cell. It takes as input the current input vector  $xt$  and the previous hidden state  $ht - 1$ , and outputs a vector  $ft$  that determines which parts of the cell state  $Ct-1$  to forget which is given in Eq. (2).

$$ft = \sigma(Wf[ht - 1, xt] + bf) \quad (2)$$

3. Memory cell: This is the key component of the LSTM architecture, which stores and updates the information over time. It takes as input the current input vector  $xt$ , the previous hidden state  $ht - 1$ , and the previous cell state  $ct - 1$ . It outputs a new cell state  $ct$ , which is a combination of the input, the forget gate, and the previous cell state which is depicted in Eq. (3).

$$ct = ft * ct - 1 + it * \tanh(Wc[ht - 1, xt] + bc) \quad (3)$$

where  $*$  is the element-wise multiplication operator,  $\tanh$  is the hyperbolic tangent activation function, and  $Wc$ ,  $Uc$ , and  $bc$  are learnable weights and biases.

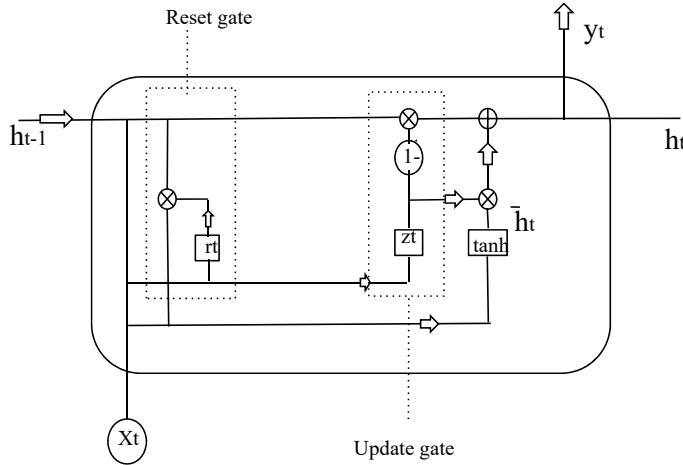
4. Output gate: This gate controls the flow of information from the memory cell to the output. It takes as input the current input vector  $xt$ , the previous hidden state  $ht - 1$ , and the current cell state  $ct$ . As shown in Eqs. (4) and (5), it generates a vector that specifies which parts of the cell state should be built as the final hidden state.

$$Ot = \sigma(Wo[ht - 1, xt] + bo) \quad (4)$$

$$ht = Ot * \tanh(Ct) \quad (5)$$

The LSTM architecture selectively updates and outputs information over time using the input gate, forget gate, and output gate, allowing it to capture long-term dependencies in sequential data.

The GRU [12] is a variant of LSTM that was developed to minimize the number of parameters and model computation time. There are two gates in the GRU: the reset gate and the update gate. The reset gate allows the model to selectively forget the previous hidden state, and the update gate allows it to selectively update the current hidden state with new data. The visual representation of the GRU is shown in Fig. 2. In mathematical terms, a GRU can be represented using two gates such as update gate and reset gate. Update gate ( $Zt$ ): This gate determines what proportion of the



**Fig. 2** GRU

previous memory state ( $h_{t-1}$ ) should be combined with the new input ( $x_t$ ) to form the new memory state ( $h_t$ ) which is given in Eq. (6).

$$Zt = \sigma(Wz[h_{t-1}, x_t]) \quad (6)$$

Reset gate ( $rt$ ): This gate determines the amount of the old memory state ( $h_{t-1}$ ) should be ignored when computing the new memory state ( $h_t$ ) given in Eq. (7).

$$rt = \sigma(Wr[h_{t-1}, x_t]) \quad (7)$$

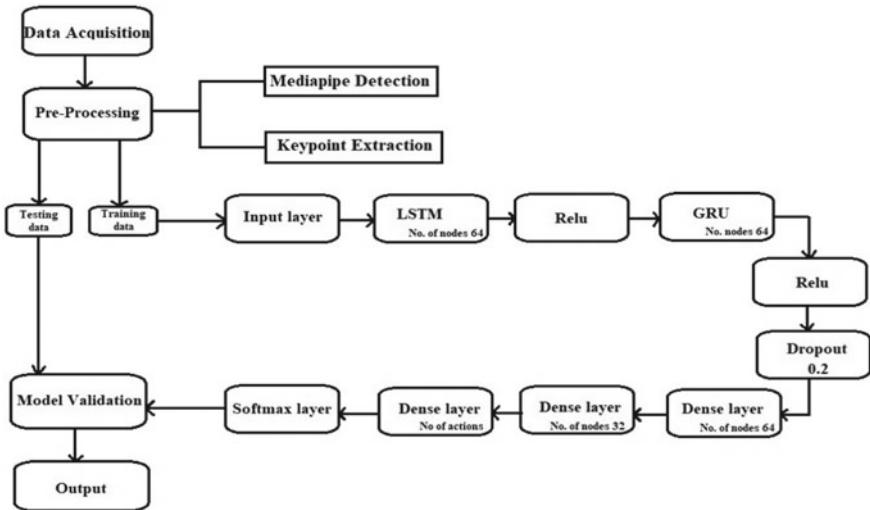
Next, we compute the candidate memory state ( $\bar{h}_t$ ), which is a combination of the new input ( $x_t$ ) and the previous memory state ( $h_{t-1}$ ) after resetting some of its elements using the reset gate given in Eq. (8).

$$\underline{h}_t = \tanh(W[rt * h_{t-1}, x_t]) \quad (8)$$

Finally, the candidate memory state ( $\underline{h}_t$ ) and the previous memory state ( $h_{t-1}$ ) are combined using the update gate ( $zt$ ) to get the new memory state ( $h_t$ ) given in Eq. (9).

$$\overline{h}_t = (1 - zt)*h_{t-1} + zt*\underline{h}_t \quad (9)$$

Firstly, a large dataset of sign language videos with spoken or written translations is collected. Mediapipe extracts hand and body key points from each gesture, allowing the calculation of features like hand movement trajectories, orientation, and body posture [13]. These features serve as inputs to the LSTM-GRU model. The LSTM-GRU model is trained through supervised learning to predict spoken



**Fig. 3** Generalized architecture of sign language interpreter

or written language translations from extracted features. Its accuracy and precision are evaluated using a webcam and specific actions. Figure 3 shows the generalized architecture of the LSTM-GRU-based sign language interpreter.

The steps followed to complete the proposed model are:

1. Initialize the Mediapipe pipeline for hand and body pose estimation.
2. Turn on the webcam to collect the dataset, sign language video sequences and their corresponding spoken or written language translations.
3. Process each video sequence by utilizing the Mediapipe pipeline to extract key point information from each frame of the sign language video, and then pair this key point information with the corresponding spoken or written language translation to create input–output pairs for the LSTM-GRU model.
4. Split the dataset into training and testing sets.
5. Train the LSTM-GRU model on the input–output pairs.
6. Evaluate the model performance on the testing set using the performance measures.
7. Fine-tune the model parameters and architecture as necessary to optimize performance.
8. Save the trained model for future use.

**Table 1** Output of the proposed LSTM-GRU model

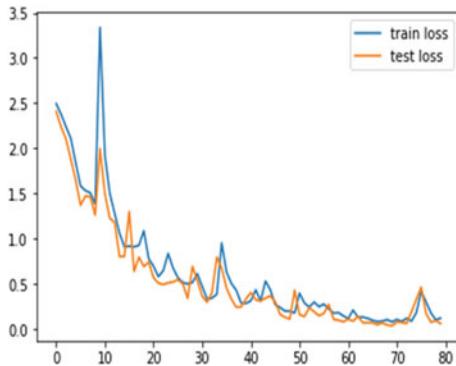
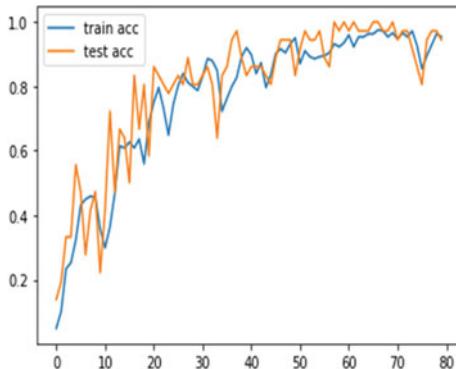
Epochs	Training accuracy of LSTM-GRU	Training loss of LSTM-GRU	Testing accuracy of LSTM-GRU	Testing loss of LSTM-GRU
10	0.313	1.549	0.413	1.564
20	0.745	0.654	0.734	0.623
30	0.846	0.451	0.857	0.412
40	0.875	0.410	0.871	0.415
50	0.913	0.421	0.942	0.184
60	0.944	0.165	0.948	0.125
70	0.965	0.120	0.952	0.091
80	0.954	0.117	0.944	0.095

### 3 Experimental Results and Discussions

Firstly, to collect the dataset, we set up a recording environment equipped with a 720p HD camera capable of capturing detailed hand movements. We enlisted a group of three participants proficient in the target sign languages. Participants were instructed to perform thirty sign language gestures while being recorded. We recorded 30 frames for each of the signs. Using media pipe detection, we extracted important key points from each of the 30 frames. The extracted key points are used to calculate features such hand movement trajectories, hand orientation, and body posture, which are fed as inputs for the LSTM-GRU model for predictions. After the model has been trained, the testing step began. The dataset is split into 10 unit using tenfold cross-validation so as to each part is used as a validation data for each epoch [14]. Training and validation will be placed concurrently, and the model is trained and validated for 80 epochs. After the training and validation phases are completed, the performance of the model is evaluated using performance metrics [15, 16]. The model accuracy and loss are defined by the number of data that is fed to the model and the number of epochs. After 80 epochs, the proposed model attained 94.4 percentage testing accuracy, and the output is shown in Table 1. The proposed LSTM-GRU model trained on a Windows 10 64-bit PC with a CPU power and 8 GB of RAM. Figures 4 and 5 depict the proposed LSTM-GRU model accuracy and loss graph, respectively. Table 2 demonstrates that the proposed prediction model, LSTM-GRU, outperformed other state-of-the-art methods in sign language interpretation.

### 4 Conclusion and Perspectives

In conclusion, the proposed model combines Mediapipe for key points extraction and the LSTM-GRU model for training and prediction. Our approach achieved promising results when it came to accurately translating sign language gestures into

**Fig. 4** Accuracy graph**Fig. 5** Loss graph**Table 2** Comparison table

Model	Train (%)	Test (%)	Recall	F1 score	Precision
Decision tree	99.89	91.52	0.91	0.91	0.91
Naïve Bayes	50.63	50.84	0.50	0.50	0.50
KNN	97.62	97.10	0.96	0.96	0.96
CNN	98.6	97.2	0.99	0.99	0.99
<b>LSTM-GRU</b>	96.8	94.4	0.96	0.96	0.96

spoken and written language. This method uses deep learning to automatically extract and classify information, resulting in a more efficient and effective method of sign language interpretation. However, there is still room for improvement in the proposed model. The wide variance in sign language movements due to individual differences, regional variations, and contextual circumstances is one of the most difficult issues in sign language interpretation. From our perspective, in the future, we could focus

on addressing these challenges by incorporating more contextual information into the model or by exploring other techniques such as transfer learning or data augmentation. Moreover, if we add more data to train our model, there is a high chance that our accuracy will increase because we know that LSTM and GRU work better with increasing dataset sizes. Overall, our proposed approach represents a promising direction for sign language interpretation and has the potential to significantly improve accessibility and inclusion for the hard-of-hearing community.

## References

1. Bilge YC, Cinbis RG, Ikizler-Cinbis N (2022) Towards zero-shot sign language recognition. *IEEE Trans Patt Anal Mach Intell* 45(1):1217–1232
2. Minu RI (2023) A extensive survey on sign language recognition methods. In: 2023 7th international conference on computing methodologies and communication (ICCMC), IEEE, pp 613–619
3. Tamiru NK, Tekeba M, Salau AO (2022) Recognition of Amharic sign language with Amharic alphabet signs using ANN and SVM. *Visual Comp*, pp 1–16
4. Akash, SK, Chakraborty D, Kaushik MM, Babu BS, Rahman Zishan, MdS (2023) Action recognition based real-time bangla sign language detection and sentence formation. In: 2023 3rd international conference on robotics, electrical and signal processing techniques (ICREST). IEEE, pp 311–315
5. Sreemathy R, Turuk M, Kulkarni I, Khurana S (2023) Sign language recognition using artificial intelligence. *Educ Inf Tech* 28(5):5259–5278
6. Aldhahri E, Aljuhani R, Alfaiadi A, Alshehri B, Alwadei H, Aljojo N, Alshutayri A, Almazroi A (2023) Arabic sign language recognition using convolutional neural network and mobilenet. *Arab J Sci Eng* 48(2):2147–2154
7. Shin J, Miah ASM, Hasan MAM, Hirooka K, Suzuki K, Lee H-S, Jang S-W (2023) Korean sign language recognition using transformer-based deep neural network. *Appl Sci* 13(5):3029
8. Nandi U, Ghorai A, Marjit Singh M, Changdar C, Bhakta S, Kumar Pal R (2023) Indian sign language alphabet recognition system using CNN with diffGrad optimizer and stochastic pooling. *Multimedia Tools Appl* 82(7):9627–9648
9. Grishchenko I, Bazarevsky V, Zanfir A, Bazavan EG, Zanfir M, Richard Yee, Raveendran K, Zhdanovich M (2022) Matthias grundmann, and cristian sminchisescu. Blazepose ghum holistic: real-time 3d human landmarks and pose estimation. arXiv preprint [arXiv:2206.11678](https://arxiv.org/abs/2206.11678)
10. Gong, S, Li M, Feng J, Wu Z, Kong LP (2022) Diffuseq: sequence to sequence text generation with diffusion models. ArXiv preprint [arXiv:2210.08933](https://arxiv.org/abs/2210.08933)
11. Goyal K (2023) Indian sign language recognition using mediapipe holistic. arXiv preprint [arXiv:2304.10256](https://arxiv.org/abs/2304.10256)
12. Kothadiya D, Bhatt C, Sapariya K, Patel K, Gil-González A-B, Corchado JM (2022) Deepsign: sign language detection and recognition using deep learning. *Electronics* 11(11):1780
13. Bora J, Dehingia S, Boruah A, Chetia AA, Gogoi D (2023) Real-time assamese sign language recognition using mediapipe and deep learning. *Proc Comput Sci* 218:1384–1393
14. Fathima MD, Hariharan R, Singh PK, Kumar P, Ramya S, Ammal MSSR (2023) Real time face mask detection using Mobilenetv2 algorithm.“ In: Recent trends in computational intelligence and its application: proceedings of the 1st international conference on recent trends in information technology and its application (ICRTITA, 22). CRC Press, p 215

15. Shewalkar A, Nyavanandi D, Ludwig SA (2019) Performance evaluation of deep neural networks applied to speech recognition: RNN, LSTM and GRU. *J Artif Intell Soft Comput Res* 9(4):235–245
16. Dhilsath FM, Justin Samuel S, Raja SP (2023) HDDSS: an enhanced heart disease decision support system using RFE-ABGNB algorithm. *Int J Interact Multimedia and Artif Intell* 527

# Can Learning Games Facilitate Open Innovation Capacity in IT Industry? The Case of Resilience



Eleni G. Makri 

**Abstract** Immersive game play for learning and development signals a regenerative successful design and learning challenge for academia, industry and software development worldwide. Inconclusive findings for gaming learning compared to conventional learning mode seem to sustain the learning gameplay “enigma” persistent. Dispersed evidence deals with open innovation-associated facets that reflect sustainability. Thereby, this study rests on exploring open innovation-related capacity in traditional and learning gameplay instruction in an IT-group organization. We report on 58 Greek employees’ open innovation-linked L&D attributes investigated after-workshop and post-Resilience gaming in 2022. The participants identified the Resilience game as more helpful instructional solution for open innovation-associated orientation in relation to conventional instruction. Male workforce and peers originated from Attica region did relate Resilience game to more open innovation-associated adaptation in their company. The obtained results are discussed in tandem with theory and practice and strands for further research that align with learning gameplay open innovation-and sustainability-orientation and experience.

**Keywords** Learning games · Open innovation · Sustainability · Industry · Greece

## 1 Introduction

The capacity to adapt to reshaping technological and organizational (e.g. workforce mobility) fusion changes across the globe, remains within collaborative (i.e. open) innovation networks. Open innovation aligns with inbound and outbound exchange of knowledge, resources, finance and technology between and across companies worldwide [1]. This inflow and outflow multimodal exchange can foster business, populations, social prosperity, environmental sustainability and resilience, and vice versa. Open innovation in the post-COVID19 pandemic era is discussed under

---

E. G. Makri (✉)  
Unicaf, Larnaca, Cyprus  
e-mail: [eleni.g.makri@gmail.com](mailto:eleni.g.makri@gmail.com)

the lens of diverse distributive collaboration, transformative co-development and cross-boundary activities (e.g. R&D, open source software, crowdsourcing, systems dynamic modelling, communities of practice) [2]. The powerful integration of immersive technologies has shifted the nature and process of production lifecycles across sectors (e.g. design, manufacturing, supply chain, IT) into challenging ones [3]. Despite shared discussions set in motion about open innovation and advanced technology challenges and opportunities, research that explores multimedia applications like the learning games for open innovation and sustainability in different infrastructure (e.g. academia and industry) seems to remain rather scarce [4]. Along this line, the present paper aims to feature the research executed as part of workshop training in open innovation and sustainability in an IT organization in Greece. On investigating open innovation and sustainability during traditional (workshop) and immersive (learning gameplay) instruction in 2022, 58 employees were assessed on their account of their open innovation and sustainability orientation after-workshop and post-gameplay instruction, as means to facilitate positive open innovation and sustainability advice through learning gaming and vice versa. Following the above rationale, therefore, the research questions that support the design of the existing study are reported as next:

- Are any differences indicated in employees' open innovation and sustainability orientation after-workshop and post-learning gaming?
- Are there any differences demonstrated in open innovation and sustainability adaptation based on legal gender (male vs female), region origin (Attica vs other), educational background (university vs technological institution) and voting status (yes vs no) after-gameplay?

## 2 Related Work

Brauner and Ziefle [3] elaborate on their single player co-designed, delivered and assessed "Quality Management Game". The authors report that they were motivated to co-develop this learning game by the rather limited focus quality management instruction adhered to production quality management within organizations. Therefore, the game aimed to support players in handling complex production within company supply chain management, in particular. The in-game environment includes the players that control the company(ies), receive supplier services and finally deliver products to the customers. Product description and costs, company profit, warehousing expenditure and payments' tasks are exchanged between company(ies) with a quality metric assigned to each product every time (i.e. well-designed/faulty). Two gamers are allocated the role of supplier and customer, respectively. Company(ies) inventories, production stage indicators, quality assurance control, and investment in new product supplies complete the rest of the game's mechanics and interface environment. Each time a product is not well-designed, of limited quality, or damaged, the customer is expected to return it. During prototyping, the authors indicate that they adjusted several parts of the in-game iteration and interface following notes and

annotations. The game ended up in consisting of three integral agents: the supplier, the manufacturer (gamer), the customer and the interrelated interaction between them. The gamers are expected to manage the in-game challenges of reflecting on the company's (ies) status by relevant key performance indicators and then act upon supplies, supply and internal quality controls. Two gamers act as supplier(s) and consumer(s). The authors assessed their game as part of a university lecture on Quality Management. 66 mechanical engineering students did play and appraised their learning game experience. The students' gameplay performance did seem to advance across different and iterative in-game levels. They perceived the game as helpful instructional tool for real-world industrial supply quality management learning challenge. They also recommended a rather limited duration of the game for better learning gameplay outcomes. However, they did perceive the game as an all-encompassing versatile instructional tool for prior dispersed supply chain and quality management learning parts. The authors further evaluated the game as part of a lab assessment on automation performance, bias and anxiety. The gamers experienced the game integrated into a decision support system that offered them inconsistent guidance on automation. Higher in-game ease of use led to easier detection of flaws, reimburse for them and accomplish greater players' trust in automation and organizational performance. Sale [5] elaborates on "Merge Defense 3D" free learning game aimed to instruct players on resource allocation supporting effective materials management in manufacturing curriculum. Resting on game mechanics and interface, the users after a number of rounds navigating across the learning game environment, they are expected to complete the corresponding learning exercise successfully. Throughout learning gameplay, the players are able to reflect on their progress on each in-game round. The production lines represented in manufacturing and distribution are depicted as tables. Small towers, keys and precious stones depict additional gaming allocation features that reflect demands to be fulfilled. The gaming challenges that need to be managed and completed effectively (e.g. budgeting, delivery deadlines) are visualized as blocks. On unsuccessful elimination of each block before each turn finishes, the game comes to an end. In this case, it tends to denote omission of a production closing date. In other ones, it might reflect missing a payment deadline. By and large, this challenge signals the overall target of the game. The learning exercise associated with the game requires players to transcribe data from the gaming interface on an Excel file. The data transcribed are expected to be translated into resource planning tools. The author reports that students did seem to be more motivated and perceive the game as positive learning experience resting on increased use of it. Post-aforementioned learning exercise completion, students' mastery goal orientation tended to be enhanced. The learners stressed the usefulness of the game to exercise concepts compared to conventional learning material and case studies. They also perceived the game as enjoyable and helpful in keeping them engaged while mastering difficult management issues. The author recommends further gameplay assessment in terms of learning performance indicators. Rauch et al. [6] report on their co-designed learning game "Additive Manufacturing". The game aims to instruct on product and procedure chain life-cycles encompassing diverse levels and aspects of sustainability. The gaming sessions span across three semesters of the

MSc in Materials Science and Engineering programme. The game is designed across three levels. The first denotes the project level where the gameplay is set up. Here the learners are assigned different roles: executive directors, process planners, and (or) production managers. The tutors take the roles of external (outbound) consultants in the game. A dynamic systems modelling depicting product and procedure design, delivery, and evaluation reflects the game mechanics and iterative interface followed. The second level involves both students and tutors interacting with each other to comprehend the gameplay environment. The third level that follows next integrates inflow and outflow iterative co-design, delivery and improvement as being co-developed, supported and implemented by all game stakeholders involved. Each game agent role adheres to one feature of diverse aspects of sustainability (technology: functions, modalities, material variety; ecology: society, economy, etc.). The in-game role of executive director addresses financial tasks, that of the procedure planner handling ecological activities, and the one of the production manager managing social challenges. The in-game conflict resolution reflects decision-making on traditional production system vs production system with additive manufacturing (or innovative one) integrated. In-game life-cycle product repairing leads to new in-game production cycles. The in-game external consultants address technology challenges in terms of viability. During semester duration, there were three gameplay cycles performed with a fourth one currently executed. The game tends to follow an ongoing iterative game mechanics and user interface improvement. Data and feedback from previous in-game cycles are used to feed into continuous in-game development. Therefore, from an almost form-free game at the beginning, the in-game mechanics and user interface became gradually more iteratively structured. However, they were not too structured at the end, as they were expected to keep the balance between a concise in-game environment and the co-development and co-implementation of new inbound and outbound game improvements. Following four in-game cycles research and development, the authors indicate the following: (a) resting on preliminary favourable gameplay feedback obtained, in-game expert advice should be available for students, (b) powder production and consumption of electrical energy should be included into the ecological sustainability nexus, (c) game supervision should be presented in a neutral way, with (d) clear definition of in-game roles allocation and sides taken during gaming. West et al. [7] present their co-designed, delivered and assessed learning game “Virtual Energy Hero”. The game received funding from the Swiss National Science Foundation. It was an R&D co-development and participatory citizen science game supported by the Zhaw School of Engineering, the Institute of Sustainable Development, the Swiss Federal Office of Energy, the City of Winterthur and Cymmersion GmbH. The game intended to instruct on energy and sustainability issues. The prototype included open call gameplay sessions ending up to 250 citizens playing the game. The iterative game mechanics and user interface were improved resting on feedback and data received from the players and user experience assessments. These assessments evaluated whether the game achieved to build an immersive learning environment that supports City of Winterthur as Smart City in terms of planning and use of urban energy, smart buildings, stakeholder engagement, effective governance, smart mobility and smart

grids/supply technologies concepts, devices and processes (p. 882). Taking a virtual learning approach in the co-design, implementation and assessment of public policy that adheres to current and future sustainable smart cities. The game builds upon the energy strategy 2050 and sustainability paradigm. The game topics exercised within the virtual Smart City of Winterthur that reflected smart city and sustainability aspects were “renewable energy, mobility, recycling, education, public transport, sustainable nutrition, and use of heating pumps” (p. 888). The players take the active role of participating in the transformation of Winterthur City as that of Smart City through gaming. Along their gameplay journey, they are supported by Oscar the owl as their advisor. Oscar acts also as their group member in entering virtual Winterthur hotspot energy and sustainability areas. The initial prototyping involved public event release six months after preliminary design. The final game was introduced after all improvement recommendations being integrated following player feedback and data from survey tools. The gaming adventure continues with the players visiting a family house owned by Oscar’s friend. The challenge is to support the family becoming more renewable energy and sustainability-driven by integrating housing improvements and taking appropriate actions. Upon completion, the players return to the balloon over the city of Winterthur where Oscar awards them with a certificate reporting their final score and motivates them to continue with learning. Survey assessment of the gameplay experience during a public hackathon indicated that 45 players perceived the game as edutainment one and obtained new knowledge. Likewise, 45 citizens reported that the game facilitated their further motivation and engagement in renewable energy topics. The game’s storyline and AR-linked iterative mechanics and user interface were perceived to be improved in the near future and integrated into an open innovation platform. The authors take a step forward recommending more future public engagement events to support environmental sustainability and resilience through emerging technologies. Reflecting on the above studies, it seems that the learning games implemented did reflect diverse aspects of open innovation and sustainability adaptation across different contexts and favourable mechanics, interface and learning experience outcomes.

### 3 Method

#### 3.1 Study Design

A total of 58 full-time employees of an IT-group organization in Greece participated in the present study as part of their onsite 3-h workshop in open innovation and sustainability. The target of the survey was to assess workforce’s attributes of open innovation after-workshop (traditional learning) and post-Resilience gaming (immersive learning), respectively. Following inconsistent evidence that compares traditional to multimodal learning [8] and limited learning game development for

open innovation-associated continuum [4], to explore whether Resilience gameplay tends to facilitate more favourable open innovation capacity when compared to usual learning, respectively. After-informed consent provided, the employees were first required to fill in demographics (i.e. gender, department/division, job role and residency items). Next, they registered for a 50-min workshop that involved open innovation-related tasks. Subsequently, they completed a 47-open innovation-oriented self-assessment continuum [9], modified for the needs of the current research to assess issues of open innovation-connected reflection and attributes exercised in their organization. The attendees answered their open innovation self-assessment based on their learning experience with the workshop instruction (post-workshop). The second and third hour of the workshop session the moderator first introduced the employees to Resilience open innovation-linked learning game through a full demo [10]. Participants were guided to experience the Resilience iterative mechanics and interface in groups [11] approximately for one hour. During and after-gaming the facilitator moderated shared reflections and debriefing feedback across participants. The employees' after-Resilience gameplay observation feedback and comments indicated were positive regarding ease of use, edutainment, and learning experience addressing the open innovation-learning continuum. The attendees were instructed to complete the matching open innovation-self-assessment after-gaming (post-gameplay).

### ***3.2 Description of the Game***

Resilience is a co-developed (i.e. open innovation product) free and award winning learning game. It has been programmed and delivered by a co-creation Drexel University USA intersectoral network (i.e. students, educators, university hubs, external refugee collaborators). The aim of the game was to build public interest and engagement in refugee sustainability and resilience [10]. Assigned the character of a refugee camp manager on a distant galaxy planet, the gamers need to tackle and resolve the challenges of extra terrestrial species invading their sphere, risking the sustainability of their camp and jeopardizing the well-being of their camp members. In succeeding to overcome the aforementioned conflicts, the players are programmed to implement open innovation-associated tasks: better comprehension of refugees' needs and motives, provision of all available internal and external stakeholder resources and tools, run shared risks, etc.). The game followed a systems dynamic modelling approach [12]. The Resilience outer space large community agents (i.e. refugees, camp manager, internal and external stakeholders, etc.) interact with each other across the learning in-game dialogue mechanics levels. The alien camp manager leads the camp on an interface tablet. The protagonists, the colours, the shapes, the buildings, the natural environment and the tablet of the camp manager on this planetary envisioned system are represented as combination of traditional earth refugee challenge reflections and outer space sci-fi imagery. Figure 1 below presents an



**Fig. 1** Screenshot of the Resilience learning game planetary camp

example screenshot of the Resilience learning game outer space camp built-up with facilities maintenance (available on <https://www.sungrazerstudio.com/press-kit>).

### 3.3 Data Analysis and Results

A total of 30 male and 28 female employees of an IT-group organization in Greece ( $N = 58$ ) from Attica ( $N = 40$ ) and other region origin ( $N = 18$ ) offered full open innovation-related account after-workshop and post-gaming instruction, respectively. In order to investigate likely differences in the experience of open innovation self-assessment post-workshop and post-Resilience gameplay, paired samples t-tests were performed as described in Table 1.

The results presented in Table 1 designate that there seem to be significant differences between all open innovation-related continuum post-workshop and after-Resilience gaming, supporting: (a) attendees' higher sense of frequency in practising open innovation by their entity and awareness of open innovation-related concepts,

**Table 1** Paired samples t-tests between open innovation post-workshop and post-Resilience gameplay ( $N = 58$ )

Open innovation self-assessment	Sig. (2-tailed)
Open innovation after-workshop-open innovation after-gaming	$t(57) = 2.10, p < 0.001^{***}$

(b) greater appreciation in applying as part of open innovation process in their organization, (c) higher insight in employing to instruct open innovation in their company, (d) greater perception in exercising for impact of open innovation in their business, (e) higher inclination for instigating as source of information for inbound innovation, and (f) superior eagerness to apply as source of information for outbound innovation, after playing the Resilience game. In sum, therefore, the Resilience learning game tends to relate employees with more open innovation-linked orientation compared to traditional learning mode.

### 3.4 2 × 2 ANOVAs

Tables 2 and 3 below demonstrate (a) the descriptives that correspond to participants' responses to the open innovation-associated continuum explored during the post-gameplay assessment by gender (i.e. male vs female), region (i.e. Attica vs other), educational background (i.e. university vs technological institution) and voting status (i.e. voting vs non-voting), and (b) further describe the main and interaction effects as to gender, region, educational background and voting for the corresponding open innovation-linked continuum assessed. Figure 2 shows the nature of the interaction effects indicated based on gender and region origin for open innovation-related account scores after-Resilience gameplay. Tables 2 and 3 results indicate that there was a significant interaction effect between gender and region origin on open innovation-related scores of participants ( $F(1,58) = 11,038, p < 0.01$ , partial  $\eta^2 = 0.170$ ), reflecting that both male and female genders and Attica and other region origin employees were influenced differently in their open innovation-associated orientation after-Resilience gameplay. In particular, the open innovation-related adaptation indicated by employees registered as male ( $M = 4.33$ ) was significantly higher than those registered as female ( $M = 4.32$ ) ( $x^2 (20) = 40,122, p < 0.001$ ), while the scores reported by Attica origin attendees ( $M = 4.37$ ) were significantly higher than those exhibited by other region origin ( $M = 4.30$ ) ( $x^2 = (20) = 35,584, p < 0.01$ ). Figure 2 depicts the nature of the interaction effect, illustrating that male gender significantly influences open innovation-related orientation for Attica region employees. Moreover, there was a significant main effect of educational background on open innovation-linked orientation ( $F(2,58) = 2,524, p < 0.10$ , partial  $\eta^2 = 0.087$ ), which denotes that university respondents scored significantly higher in their open innovation-related capacity than their technological institution peers post-Resilience gaming ( $M = 4.49$  and  $M = 4.34$ , accordingly, estimated marginal means). Further, there was a significant main effect of voting status on open innovation-related agency ( $F(1,58) = 6,458, p < 0.05$ , partial  $\eta^2 = 0.109$ ), which indicates that non-voting employees scored significantly higher in their open innovation-associated capacity than their voting colleagues after-Resilience gaming ( $M = 4.41$  and  $M = 4.34$ , respectively, estimated marginal means). However, the aforementioned main effects of educational background and voting status were not validated by a significant interaction effect between educational background and

**Table 2** Comparisons between the participants' post-Resilience gaming responses related to open innovation-associated continuum separately after controlling for gender and region origin

Open innovation-related continuum	Gender (M, (MM))		Region origin (M, (MM))		2-way ANOVA between gender and region origin ( $\alpha = 0.05$ )
Open innovation (OI) (47 items, $\alpha = 0.80$ )	Male	Female	Attica	Other	Main effect analysis: no significant difference between male and female (p almost equals to 0.1) and between Attica and other origin (p almost equals to 0.6). Interaction effect analysis: $F(1,58) = 11,038, p < 0.01$ , partial $\eta^2 = 0.170$ . Male higher than female ( $\chi^2 (20) = 40,122, p < 0.001$ ) and Attica origin higher than other ( $\chi^2 = (20) = 35,584, p < 0.01$ ) (Fig. 2)
	4.33 (4.34)	4.32 (4.27)	4.37 (4.32)	4.30 (4.30)	

Notes Male N = 30; Female N = 28; Attica N = 40; Other N = 18; Total N = 58;  $\alpha$  = Cronbach's Alpha, M = Mean, MM = Estimated Marginal Mean,  $\alpha$  = the limit of the significant level

voting status on open innovation-orientation post-Resilience gaming, reflecting that university and technological institution employees as well as voting and non-voting participants were not affected differently in their open innovation-associated capacity after-Resilience gameplay. In sum, therefore, the Resilience game seems to connect more male and Attica region origin employees with open innovation-related capacity. Figure 3 below outlines the conceptual model that reflects the revealed post-gaming results, as next.

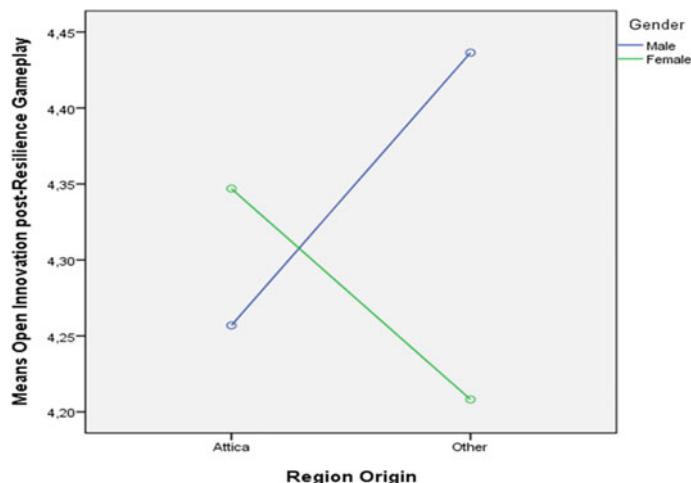
## 4 Discussion

The existing study aims to be inventive by embedding and exploring two enduring learning and development challenging issues: the open innovation-associated account mirroring sustainability capacity and the instruction mode (conventional/workshop vs learning game), respectively. In other words, the open innovation-related agency investigated after-workshop and post-Resilience learning gaming in an IT industry, accordingly. On the whole, the indicated evidence does seem to associate Resilience learning game instruction with more positive employee open innovation-connected orientation compared to workshop guidance. That is, with more favourable workforce adaptation to implicit and explicit elements of open innovation alignment post-Resilience learning gaming that relate to: (a) operationalization and exercise of open innovation, (b) practice of open innovation process (e.g. through approaches or sectors like systems dynamic modelling, technology innovation, business intelligence, product or service development, manufacturing and distribution), (c) instruction of open innovation concept and exercise (including e.g. reflection on customers' needs and motives, new technology, accelerate time for market, risk sharing in

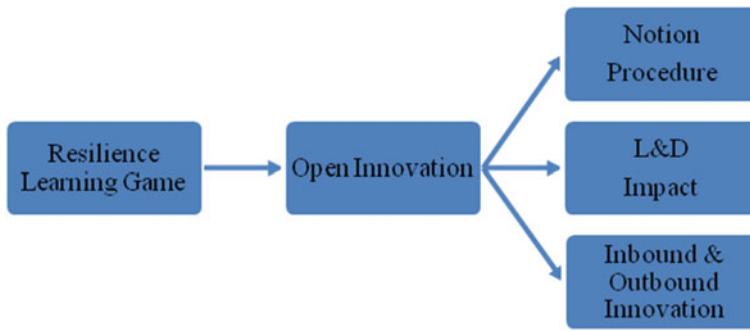
**Table 3** Comparisons between the participants' post-Resilience gaming reactions related to open innovation-associated continuum separately after controlling for educational background and voting status

Open innovation-related continuum	Educational background ( $M$ , (MM))		Voting status ( $M$ , (MM))		2-way ANOVA between educational background and voting status ( $\alpha = 0.05$ )
Open innovation (OI) (47 items, $\alpha = 0.80$ )	U	TI	Voting	Non-voting	$F(2,58) = 2,524, p < 0.10$ , partial $\eta^2 = 0.087$ (educational background). $F(1,58) = 6,458, p < 0.05$ , partial $\eta^2 = 0.109$ (voting status). Interaction effect analysis: no significant difference between educational background and voting status ( $p$ almost equals to 0.1)
	4.48 (4.49)	4.29 (4.34)	4.30 (4.34)	4.39 (4.41)	

Notes U = University; TI = Technological Institution; Male N = 30; Female N = 28; Attica N = 40; Other N = 18; Total N = 58;  $\alpha$  = Cronbach's Alpha, M = Mean, MM = Estimated Marginal Mean,  $\alpha$  = the limit of the significant level



**Fig. 2** Interaction effect for open innovation self-assessment after-Resilience gaming by gender and region origin



**Fig. 3** Outline of conceptual model indicating Resilience after-gaming open innovation-learning continuum

complex infrastructure, support of supplier network, open data and sharing, IP management advice). Also, (d) in success of open innovation in their company for support of middle and top management, project management, resource allocation, L&D, (e) as source of information for inflow innovation for internal resources, within their organization and its' subsidiary companies, software developers, business service providers, within academic, consulting, R&D, communities of practice, public service, industry and government agencies and (f) as source of information for outflow innovation for knowledge or technology transfer to external partners. The aforementioned findings lend support to previous ones designed to facilitate sustainability with more engaging game mechanics and interface experience along with successful learning outcomes (e.g. purchasing and supply management: [13, 13] compared to traditional in-class instruction). In addition, they tend to extend prior ones that reflect open innovation specific facets (e.g. [15]: portfolio of idea assessment in collaborative innovation context), by exploring a rather more holistic open innovation-related adaptation account (i.e. practice and tools for open innovation resting on inbound and outbound innovation: [9, 9]) in workforce Resilience learning gameplay. Furthermore, the present results seem to expand (a) prior ones that associate learning gaming with university students' entrepreneurship attributes (e.g. Entrexplorer game: [17]) and b) corroborate and extend previous ones reflecting male legal genders related more to global citizenship-associated sustainability [18] to industry learning gameplay sustainability orientation. Likewise, they seem to be innovative in connecting open innovation-related orientation with individual differences domain (i.e. legal gender, region origin, educational background and voting status) across in-learning gaming workforce instruction or else, exploring individual differences within immersive work-based L&D. By and large, we seem to perceive our work as reflecting theoretical and empirical contribution towards (a) expanding conventional entrepreneurship and innovation-related research in higher education and industry to open innovation-associated one explored across multimedia learning gameplay, (b) offering insights into the connection between open innovation and immersive learning gaming in IT industry cross-culturally (i.e. Greece) and (c)

linking rich learning gameplay at work with open innovation-related adaptation driving sustainability. By promoting the favourable open innovation-associated and sustainability-reflected aspects of Resilience learning game, we seek to encourage further the co-development and exploration of open innovation and sustainability learning gameplay in academic and industrial domains. This future investigation would benefit by focusing on open innovation and sustainability particulars as well as by taking into consideration individualized learning gameplay compared to classroom instruction in academia and industry. In this respect, therefore, it might further support open innovation and sustainability immersive solutions.

## 5 Conclusion

After-workshop compared to post-Resilience learning gaming instruction, IT-group organization employees identified the co-developed game as more positive open innovation and sustainability-related tool. In addition, male attendees and the ones originated from Attica did indicate more promising open innovation and sustainability-reflected orientation after-Resilience gameplay. In this regard, therefore, this evidence might be promising for further development of co-designed and multimodal learning games for integrated curricula.

**Acknowledgements** The author appreciates the assistance and collaboration offered by the IT-group organization and the time and effort that participants assigned in making this research possible.

In loving memory of my parents, doctor Georgios Makris, teacher Georgia Tsiotou-Makri and my friend Dolin. Thank you for your kind concern and collaboration.

## References

1. Eradatifam M, Heydarabadi S, Shahbazi A (2020) The impact of design thinking on innovation. *J Des Think* 1(1):49–60. <https://doi.org/10.22059/jdt.2020.76036>
2. Beck S, Bergenholz C, Bogers M, Brasseur TM, Conradsen ML, Di Marco D, Distel AP, Dobusch L, Dörler D, Effert A, Fecher B, Filiou D, Frederiksen L, Gillier T, Grimpe C, Gruber M, Haeussler C, Heigl F, Hoisl K, Hyslop K, Kokshagina O, LaFlamme M, Lawson C, Lifshitz-Assaf H, Lukas W, Nordberg M, Norn MT, Poetz M, Ponti M, Pruschak G, PujolPriego L, Radziwon A, Rafner J, Romanova G, Ruser A, Sauermann H, Shah SK, Sherson JF, Suess-Reyes J, Tucci CL, Tuertscher P, Vedel JB, Velden T, Verganti R, Wareham J, Wiggins A, Mosangzi Xu S (2022) The open innovation in science research field: a collaborative conceptualisation approach. *Ind Innov* 29(2):136–185. <https://doi.org/10.1080/13662716.2020.1792274>
3. Brauner P, Ziefle M (2022) Beyond playful learning—serious games for the human-centric digital transformation of production and a design process model. *Technol Soc* 71:102140. <https://doi.org/10.1016/j.techsoc.2022.102140>
4. Feldmann N, Adam MTP, Bauer M (2014) Using serious games for idea assessment in service innovation. In: 22nd European conference on information systems. Tel Aviv, Israel, pp 1–17
5. Sale SR (2022) The game of teaching resource allocation. *Int J Edu Dev ICT* 18(2):215–222. <http://ijedict.dec.uwi.edu/viewarticle.php?id=2992>

6. Rauch C, Maurer O, Lang SE, Bähre D (2023) Serious games in academic education—a multi-dimensional sustainability analysis of additive versus conventional manufacturing technologies in a fictitious enterprise project. In Kohl H, Seliger G, Dietrich F (eds) Manufacturing driving circular economy. GCSM 2022. Lecture Notes in Mechanical Engineering. Springer, Cham. [https://doi.org/10.1007/978-3-031-28839-5\\_91](https://doi.org/10.1007/978-3-031-28839-5_91)
7. West M, Yildirim O, Harte AE, Ramram A, Fleury NW, Carabias V (2019) Enhancing citizen participation through serious games in virtual reality. In: Schrenk M, Popovich VV, Zeile P, Elisei P, Beyer C, Ryser J (eds) REAL CORP 2019 Proceedings, pp 881–888. <https://doi.org/10.48494/REALCORP2019.2125>
8. Salim H, Stewart RA, Sahin O, Sagstad B, Dudley M (2021) R3SOLVE: a serious game to support end-of-life rooftop solar panel waste management. *Sustainability* 3(22):12418. <https://doi.org/10.3390/su132212418>
9. Cosh A, Zhang JJ, Bullock A, Milner I (2011) Open innovation choices—what is british enterprise doing? Centre for business research and UK-IRC. University of Cambridge
10. Resilience Homepage, <https://sungrazerstudio.itch.io/resilience>. Accessed on 30 06 2022
11. Caserman P, Hoffmann K, Müller P, Schaub M, Straßburg K, Wiemeyer J, Bruder R, Göbel S (2020) Quality criteria for serious games: serious part, game part, and balance. *JMIR Ser Games* 8(3):e19037. <https://doi.org/10.2196/19037>
12. Resilience Development Team Homepage. <https://www.sungrazerstudio.com/home>, accessed on 01 June 2022
13. Lauben L, Roszko J, Gallegos A, Perry Z (2022) Building resilience: multidisciplinary research, iterative processes, and serious game design. *J Games, Self Soc* 3(1):9–42. <https://doi.org/10.1184/R1/12215417>
14. Delke V, Schiele H, Buchholz W (2021) Assessing serious games within purchasing and supply management education: an in-class experiment. In: 15th European conference on games based learning (ECGBL). Brighton, UK, pp 178–189. <https://doi.org/10.34190/GBL.21.143>
15. Hauge JB, Duin HK, Thoben D (2008) Applying serious games for supporting idea generation in collaborative innovation processes. In: 2008 IEEE international technology management conference (ICTE). Lisbon, Portugal, pp 1–8
16. Chesbrough H, Bogers M (2014) Explicating open innovation: clarifying an emerging paradigm for understanding innovation. In: Chesbrough H, Vanhaverbeke W, West J (eds) New frontiers in new innovation. Oxford University Press, pp. 3–28
17. Almeida FLF (2017) Learning entrepreneurship with serious games. a classroom approach. *Int Edu Appl Sci Res J* 2(1):1–6. <https://doi.org/10.48550/arXiv.1710.04118>
18. Kuhn HP (2010) International perspectives on political socialization and gender: an introduction. In: Ittel A, Merkens H, Stecher L, Zinnecker J (eds) *Jahrbuch Jugendforschung* 8 2008/2009. VS Verlag für Sozialwissenschaften, Wiesbaden, pp 11–24

# Investigating Role of SVM, Decision Tree, KNN, ANN in Classification of Diabetic Patient Dataset



Sarita Kumari and Amrita Upadhyaya

**Abstract** Diabetes, which is a long-term ailment, is characterized primarily by high levels of sugar in the blood. It has been connected to a broad range of different types of complicated disorders, such as heart attack, renal failure, and stroke, among others. Almost 422 million people throughout the world were diagnosed with diabetes in 2014, and according to IDF Atlas 2021 report, 10.5% of the adult population (20–79 years) has diabetes, by 2045, IDF projections show that 1 in 8 adults approx. 783 million will be living with that disease an increment to 46% making it the most common metabolic. Logistic regression was used in traditional research to determine the characteristics that increase a person's likelihood of developing diabetes based on probability value and odds ratio. The authors utilize many classifiers to make predictions about diabetes patients, including NB, DT, AB, and RF. Twenty separate tests were conducted, each using one of three partitioning strategies. These classifier's effectiveness is measured by their accuracy and area under the curve. The overall accuracy rate of ML systems was 90.62% with conventional research. The K10 procedure combined LR-based feature selection with an RF-based classifier to obtain an ACC of 94.25% and an AUC of 0.95. The major goal of this work is to compare the performance of SVM, Decision Tree, KNN, & ANN on a dataset of diabetes patient classifications. The study's goals include improved accuracy and trustworthy results.

**Keywords** Diabetic patient · SVM · Decision tree · LR · KNN · ANN · Gaussian NB

---

S. Kumari (✉) · A. Upadhyaya

Department of Computer Science and Engineering, Banasthali Vidiya Peeth, Rajasthan, India  
e-mail: [saritanaveenkaliraman@gmail.com](mailto:saritanaveenkaliraman@gmail.com)

## 1 Introduction

### 1.1 Diabetes

If someone has diabetes, then the body will always struggle to use the food you eat as fuel. Sugar from the foods you consume is mostly taken into your circulation as glucose [1]. The pancreas releases insulin in response to an increase in blood glucose levels [2]. Insulin is a crucial hormone that facilitates glucose entry into cells, where it can be used as fuel. Diabetics either fail to produce enough insulin or fail to effectively use the insulin they do produce [3]. Excess glucose stays in the blood when cells stop responding to insulin or when insulin synthesis is insufficient. Illnesses such as heart disease and kidney failure may develop over time [4].

### 1.2 Types of Diabetes

It's worth noting that there are many distinct kinds of diabetes [5]. In general, you may classify them as one of these:

- **Type 1 Diabetes:** It is thought that an immune response leads to type 1 diabetes. This response leads to a decrease in insulin secretion. Only about 5–10% of diabetics are affected by type 1. Most people with type 1 diabetes have a rapid onset of their symptoms. Young people (kids, teenagers, and 20-somethings) are the typical patients [6]. Survival requires daily insulin injections for those with type 1 diabetes. No one has yet discovered a way to stop the onset of type 1 diabetes.
- **Type 2 Diabetes:** High blood sugar levels persist despite regular use of insulin in those with type 2 diabetes. Around 90–95% of all cases of diabetes are type 2. It takes years to develop; therefore, adults are the ones who often get a diagnosis. If we are at risk, they should have to check blood sugar even if they don't feel sick [7]. Changes in diet and exercise routines, as well as other healthy habits, may help prevent or postpone the onset of type 2 diabetes.
- **Gestational (Type 3) diabetes:** Hyperglycemia during pregnancy, often known as gestational diabetes, is characterized by elevated blood glucose levels that fall short of diagnosing diabetes. In most cases, gestational diabetes manifests itself during pregnancy. As a result, women with gestational diabetes are more likely to have difficulties throughout pregnancy and delivery [8]. The likelihood that these moms, and perhaps their children, would acquire type 2 diabetes is increased. Prenatal screening for risk factors is the gold standard for identifying gestational diabetes, rather than patient self-reporting.

Using all of this medical information, we built the machine learning model and used data analysis and visualization to draw some conclusions.

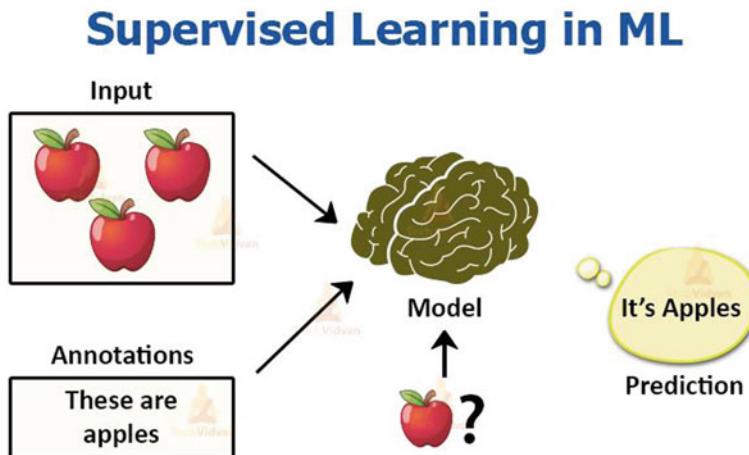
### 1.3 Machine Learning

Machine learning enables systems to make choices independently and without outside assistance. When the computer can comprehend the underlying patterns in the data and learn from it, it makes these choices [9]. They then yield the result, which can be a classification or a prediction, through pattern matching and additional analysis. Machine learning types are:

**Supervised learning:** The most common paradigm for carrying out machine learning tasks is supervised learning (as shown in Fig. 1). It is often employed for data in which the input–output data are precisely mapped. The methodology used by the AI system to carry out its task—generally, predicting results from given input statistics is known as a machine learning algorithm [10]. Classification and regression are the two primary methods applied by machine learning algorithms.

**Unsupervised Learning:** In unsupervised learning, the computer is left to draw its own conclusions based on the information it has been fed. The programmer must process the data autonomously, and the unstructured models can be taught with the unlabeled dataset because it lacks classifications [12]. In unsupervised learning, the model does not attempt to produce a preset outcome but rather looks through the massive quantity of data for useful insights. To solve the Association and Clustering problems, they are used.

**Reinforcement Learning:** By taking in the results of its activities and adapting accordingly, an entity engages in reinforcement learning. The entity gets reinforcement in the form of incentives, such as positive reinforcement for appropriate



**Fig. 1** Supervised learning [11]

behavior and negative reinforcement for inappropriate behavior [13]. The representative is not supervised in any way. The Q-learning method is used in reinforcement learning.

## 1.4 SVM

It can be applied to both categorization and regression problem, SVM is a popular guided learning technique. However, in the realm of machine learning, its most frequent application is in classification jobs [14]. For future reliable classification of new data points, the SVM approach is designed to provide the optimum line or judgment border that can divide n-dimensional space into groups. Within the boundaries of a hyperplane [15], the greatest choices may be made. To this end, the SVM might take the form of either.

- **Linear SVM:** It is possible to apply a classifier called a linear SVM when a dataset is linearly separable, which means that a straight line is sufficient to split it into two parts.
- **Non-linear SVM:** To identify datasets that do not conveniently fit into a linear hierarchy; we use a classifier called a nonlinear SVM.

## 1.5 Decision Tree

A Decision Tree is a paradigm for categorizing data that takes the form of a tree. A Decision Tree breaks down the information into branches that might be other trees or just simple nodes [16]. The nodes in a Decision Tree might be one of three kinds.

- **Decision Nodes:** In this case, the node has two or more offshoots.
- **Leaf Nodes:** The deciding nodes at the bottom of the tree.
- **Root Node:** At the highest level, this is another decision node.

**Types of Decision Trees:** Decision Tree algorithms owe a great deal to Hunt's algorithm, which was created in the 1960s to simulate the way people learn in psychology.

**ID3:** ID3 which stands for “Iterative Dichotomizer 3” is a system that was created by Ross Quinlan. This technique uses the entropy and information gain to rank possible partitions [17].

**C4.5:** This method is a refinement of ID3, which Quinlan also created. Decision tree forks may be ranked according to information gain or gain ratios.

**CART:** CART stands for “classification and regression trees” an acronym coined by Leo Breiman. The Gini impurity is often used by this approach to choose the most suitable splitting property [17].

## 1.6 *K-Nearest Neighbors*

Nonparametric slow supervised learning method NN is often used for classification issues. There's a lot to dissect, but here are the two most important features of the KNN:

- To further clarify, KNN is a nonparametric algorithm, which means that the model does not assume anything about the distribution of underlying data.
- As a lazy learner method, KNN does not use the training dataset to teach the algorithm how to discriminate between classes [18]. To avoid this, a lazy learner algorithm memorizes the training dataset and waits to abstract the data until it is requested to make a prediction.

## 1.7 *Artificial Neural Network*

The term “artificial neural network” was coined to mimic the functioning of the biological neural networks responsible for shaping the human brain throughout embryonic stages [19]. Layers of “neurons” are connected to one another in ANN in the same way as neurons in the brain are connected to one another. These groups of neurons are called nodes.

## 2 Literature Review

The findings of healthcare datasets analyzed and predicted using different approaches are presented in the reviewed relevant literature. Researchers have created and applied a wide variety of forecast models, often utilizing hybrids of data mining and machine learning approaches.

Noi et al. [1] focused on the k-nearest neighbor, SVM, Land Cover Classification in Sentinel-2 Satellite Imagery using SVM and RF classifiers. High precision was attained both for skewed and balanced datasets.

Alehegn et al. [2] introduced type 2 diabetes ensemble approaches. This study's methodology makes use of PIDD and 130 US hospital diabetes datasets.

Mahabub et al. [3] looked at potential complications from diabetes using an established and reliable ML voting technique. The purpose of this article was to present a thorough list of ML and DM uses in the context of diabetes.

Sharmila Agnai et al. [4] did research on the naïve Bayes classifier for evaluating diabetes data. The graph indicates that both the incidence and prevalence of diabetes have been increasing over the last several decades.

Afrianto et al. [5] logistics regression, DT, KNN, and RF Classifier as Booking Prediction Models for FSBO Accommodation Listings. Based on the findings,

Random Forest Classifiers had the highest area under the operating curve of all models analyzed (AUC-ROC).

Zhu et al. [6] presented work on phishing detection using a DTOF-ANN. To address this deficiency, this article develops a DTOF-ANN, neural network phishing detection model that makes use of Decision Trees and selects features in the optimum way.

Alghurair et al. [7] reviewed SVM, ANN, LGBM, and LR may all benefit from generic frameworks. The given frameworks were compared to other state-of-the-art alternatives, with the second solution emerging as the clear winner.

Maniruzzaman et al. [8] looked ML paradigm for illness classification and prediction in diabetes. The diabetic patient predictions were made using a combination of four classifiers, which were used to assess the classifiers' overall effectiveness.

Pathak et al. [9] presented work on investigated IDS using DT and the KNN algorithm. In this study, they implement and assess the accuracy of two ML approaches, the Decision Tree and the KNN, using IDS.

Shah et al. [10] looked at LR, RF, and KNN are the three models for text categorization. In this research, they create a system for labeling BBC News stories.

Jijo et al. [12] reviewed the DT algorithm for ML classification. When it comes to Decision Trees, this paper has you covered. Specifics of the study were also examined and discussed in detail, including the algorithms/approaches used, the datasets utilized, and the results obtained.

Mohideen et al. [13] researched the accurate prediction of diabetes mellitus, they combined a Gaussian naive Bayes algorithm with regression imputation.

### 3 Problem Statement

Many studies have been conducted in the field of diabetes patient data categorization. Traditional studies, however, have a problem with space and time consumption owing to huge-sized pictures. Prediction of a DM diagnosis using patient-reported characteristics was accomplished using SVM. Without diabetes, at risk for developing diabetes, and diagnosed with diabetes are the three possible values for the outcome variable. Classification techniques are utilized to foretell cases of diabetes. In this research, we use and evaluate many different methods for diabetes prediction, with many different hidden layers.

### 4 Dataset Used

In this review paper, we work on how to gather and analyze data to identify patterns and trends that can then be used for future forecasting and product evaluation. A summary of the dataset is provided below. In Table 1, some relevant papers that

**Table 1** Dataset information

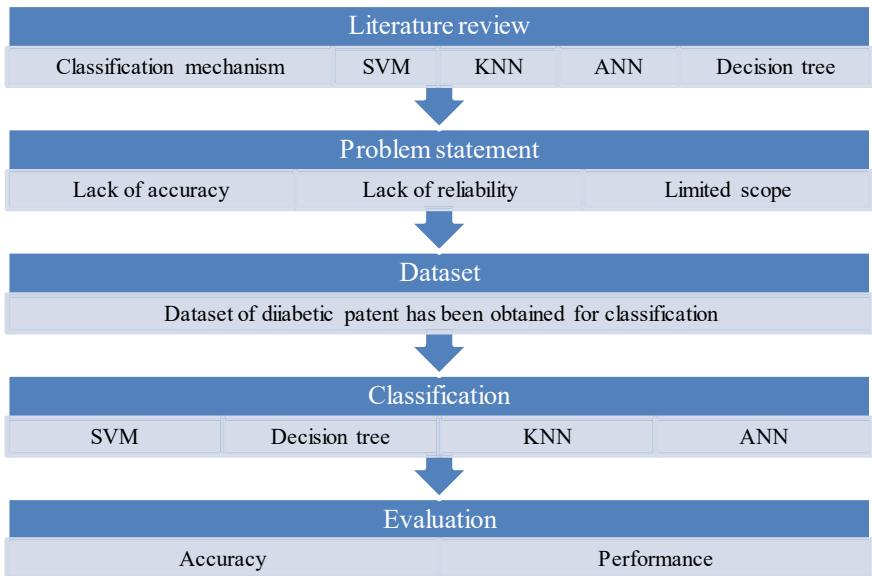
Name	Description	Format
Diabetes patients data [20]	Most of this data comes from studies funded by the National Institute of Diabetes and Digestive and Kidney Diseases	From the dataset in the (.csv) File, and can find several variables
N. Inst. of diabetes and kidney dis [21]	Most of this data comes from studies funded by the National Institute of diabetes and digestive and kidney diseases. The purpose of these studies is to confirm the presence of diabetes in a patient	From the dataset in the (.csv) File
Diabetes health indicators dataset [22]	The CDC collects the BRFSS every year, and it's a health-related telephone poll	Diabetes _ 012 _ health _ indicators_ BRFSS2015.csv is a clean dataset of 253,680 survey responses
Diabetes disease updated dataset [23]	Most of this data comes from studies funded by the national institute of diabetes and digestive and kidney diseases	From the dataset in the (.csv) File
Diabetes dataset for beginners [24]	Multiple medical predictive factors and a single outcome variable make up each dataset	Dataset in (.csv) File

reflect past work in this field are taken as base papers for this review paper, and we have to give a better analysis form these papers.

## 5 Proposed Research Methodology

There the research methodology used in research has been discussed. There have been several conventional mechanisms that focused on the classification of data. Present research work has investigated the role of SVM, Decision Tree, KNN, and ANN in the classification of diabetic patient datasets. Research work has considered conventional classification techniques. Simulation of accuracy has been made over a diabetic patient dataset using SVM, Decision Tree, KNN, and ANN to evaluate the reliability of classifiers [25, 26].

Research work has compared the accuracy of ANN to conventional classification mechanisms such as SVM, Decision Tree, and KNN to get better accuracy. Figure 2 presents the process flow of the training dataset in the ANN model and getting accuracy. All the steps we followed are necessary and help us to resolve the points we want to make.



**Fig. 2** Flow of work in this paper

## 6 Comparison of Accuracy of Conventional ML Approaches

The model's overall success is summarized and presented in the shape of a matrix. The following are the results we obtained after employing several different ML algorithms in the dataset. With 96% precision, logistic regression is the most accurate method. Table 2 presents the accuracy in the case of different classifiers.

Table 3 presents the outcome of the simulation in the case of 4 mechanisms that are used for the classification of the diabetic dataset.

As shown in Table 3 there is a comparison between many classifiers, and all are done on the same that is diabetic datasets. There the comparison done between SVM, Decision Tree, KNN, and ANN all have their accuracy levels that are 84, 86, 90, and 96%. According their accuracy, ANN is giving better results as compared with other classifiers.

In Fig. 3, comparison of accuracy for different classification mechanisms is shown. The given chart briefly describes all the classifier's accuracy in proper and accurate. This may include Table 3 in entries to represent them correctly graph.

Table 4 is presenting the comparison of the error rate for the classification technique after finding the best accuracy in Table 3.

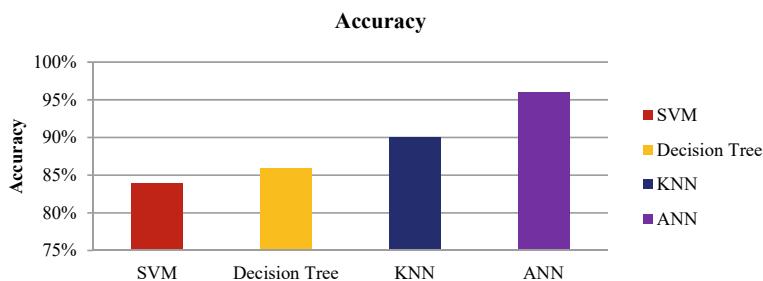
As shown in Table 4, there is a comparison between many classifiers and all are done on the same that is diabetic datasets. The comparison is done between SVM,

**Table 2** Accuracy table

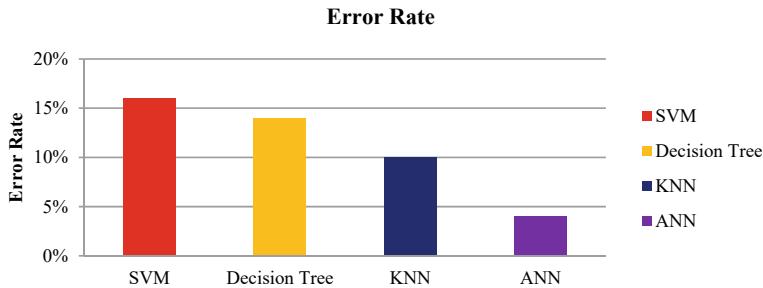
Algorithms	Accuracy (%)
Decision Tree [6]	86
Gaussian NB [13]	93
LDA [19]	94
SVC [19]	60
Random forest [1]	91
Extra trees [14]	91
AdaBoost [19]	93
Perceptron[19]	76
Logistic regression [5]	96
Gradient boost classifier [19]	93
Bagging [19]	90
KNN [2]	90

**Table 3** Comparison of accuracy for different classification mechanisms

Mechanism	Accuracy (%)
SVM	84
Decision Tree	86
KNN	90
ANN	96

**Fig. 3** Comparison of accuracy for different classification mechanisms**Table 4** Comparison of error rate for different algorithms

Algorithm	Accuracy (%)
SVM	16
Decision Tree	14
KNN	10
ANN	4



**Fig. 4** Comparison of error rate for classification technique

Decision Tree, KNN, and ANN all have error rates that are 16, 14, 10, and 4%. ANN has a minimum error rate, and SVM has the highest error rate which is 4 and 16%.

Research shows the error rate of all the classifiers in tabular form and generates its corresponding chart too Fig. 4. This may include Table 4 in entries to represent them correctly graph.

## 7 Conclusion

Considering a review of conventional approaches used for classification. It is concluded that accuracy in the case of SVC and perceptions is minimum which is 60 and 76%, whereas Decision Tree provides accuracy of 86%. Bagging and KNN are providing an accuracy of 90%. 91% accuracy has been provided by Random Forest and Extra Trees. The accuracy of Ada Boost, Gaussian NB, and Gradient Boost Classifier is the same which is 93%. High-test accuracy has been provided by logistic regression and LDA which is 96% and 94%. In the present simulation, it has been concluded that ANN has provided better accuracy as compared to SVM, Decision Tree, and KNN.

## 8 Future Scope

When data points cannot be partitioned along a linear axis, SVM uses a high-dimensional feature space to categorize them. After determining where the two groups part ways, the information is transformed such that the boundary may be shown visually as a hyperplane. To circumvent the challenges of employing linear functions in high-dimensional feature space, SVM and support vector regression may be employed to change the optimization issue into dual-convex quadratic approaches.

## References

1. Thanh Noi P, Kappas M (2017) Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. Sensors (Basel) 18(1). <https://doi.org/10.3390/s18010018>
2. Alehegn M, Joshi RR, Mulay P (2019) Diabetes analysis and prediction using random forest, KNN, Naïve Bayes, and J48: an ensemble approach. Int J Sci Technol Res 8(9):1346–1354
3. Mahabub (2019) A robust voting approach for diabetes prediction using traditional machine learning techniques. SN Appl Sci 1(12):1–12. <https://doi.org/10.1007/s42452-019-1759-7>
4. Agnai S, Saraswathi E (2020) Analyzing diabetic data using naive-bayes classifier. Eur J Mol Clin Med 7(4):2687–2698
5. Afrianto MA, Wasesa M (2020) Booking prediction models for peer-to-peer accommodation listings using logistics regression, decision tree, K-nearest neighbor, and random forest classifiers. J Inf Syst Eng Bus Intell 6(2):123. <https://doi.org/10.20473/jisebi.6.2.123-132>
6. Zhu E, Ju Y, Chen Z, Liu F, Fang X (2020) DTOF-ANN: an artificial neural network phishing detection model based on decision tree and optimal features. Appl Soft Comput J 95:106505. <https://doi.org/10.1016/j.asoc.2020.106505>
7. Alghurair NI, Mezher MA (2020) Generic frameworks for Svm, Ann, Lgbm, and Lr Algorithms. Int J Comput Sci Mob Comput 9(6):132–140. [Online]. Available: <https://www.academia.edu/download/63787039/V9I6202035.pdf>
8. Maniruzzaman M, Rahman MJ, Ahammed B, Abedin MM (2020) Classification and prediction of diabetes disease using machine learning paradigm. Heal. Inf. Sci. Syst. 8(1):1–14. <https://doi.org/10.1007/s13755-019-0095-z>
9. Pathak A, Pathak S (2020) Study on decision tree and KNN algorithm for intrusion detection system. Int J Eng Res V9(05):376–381. <https://doi.org/10.17577/ijertv9is050303>
10. Shah K, Patel H, Sanghvi D, Shah M (2020) A comparative analysis of logistic regression, random forest and KNN models for the text classification. Augment Hum Res 5(1). <https://doi.org/10.1007/s41133-020-00032-0>
11. <https://techvidvan.com/tutorials/wp-content/uploads/sites/2/2020/07/Supervised-Learning-in-ML.jpg>
12. Charbuty, Abdulazeez A (2021) Classification based on decision tree algorithm for machine learning. J Appl Sci Technol Trends 2(01):20–28. <https://doi.org/10.38094/jast20165>
13. Mohideen FM, Raj JSS, Raj RSP (2021) Regression imputation and optimized gaussian naive bayes algorithm for an enhanced diabetes mellitus prediction model. Brazilian Arch Biol Technol 64. <https://doi.org/10.1590/1678-4324-2021210181>
14. Kiranashree BK, Ambika V, Radhika AD (2021) Analysis on machine learning techniques for stress detection among employees. Asian J Comput Sci Technol 10(1):35–37. <https://doi.org/10.51983/ajcst-2021.10.1.2698>
15. Noori NA, Yassin AA (2021) A comparative analysis for diabetic prediction based on machine learning techniques. J Basrah Res 47(1):180–190
16. Rahman P, Rifat A, Chy IA, Khan MM, Masud M, Aljahdali S (2022) Machine learning and artificial neural network for predicting heart failure risk. Comput Syst Eng 44(1):757–775. <https://doi.org/10.32604/csse.2023.021469>
17. Bansal M, Goyal A, Choudhary A (2022) A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning. Decis Anal J 3(November 2021):100071. <https://doi.org/10.1016/j.dajour.2022.100071>
18. Almutairi S, Abbod MF (2023) Machine learning methods for diabetes prevalence classification in Saudi Arabia. Modelling 4(1):37–55. <https://doi.org/10.3390/modelling4010004>
19. Mujumdar, Vaidehi V (2019) Diabetes prediction using machine learning algorithms. Proc Comput Sci 165:292–299. <https://doi.org/10.1016/j.procs.2020.01.047>
20. <https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset>
21. <https://www.kaggle.com/datasets/mathchi/diabetes-data-set>

22. <https://www.kaggle.com/datasets/alextreboul/diabetes-health-indicators-dataset>
23. <https://www.kaggle.com/datasets/jillanisofttech/diabetes-disease-updated-dataset>
24. <https://www.kaggle.com/datasets/shantanudhakadd/diabetes-dataset-for-beginners>
25. <https://www.kaggle.com/datasets/alakaay/diabetes-uci-dataset>
26. <https://idf.org/about-diabetes/facts-figures/>

# Optimal Resource Allocation in Cloud Computing Using Novel ACO-DE Algorithm



Himanshu Bhushan Sahoo and D. Chandrasekhar Rao

**Abstract** Cloud computing has emerged as a popular paradigm for delivering on-demand computing resources over the Internet. Resource allocation and load balancing are crucial elements of job scheduling because they enable the distribution of a growing volume of jobs to a finite number of virtual machines (VMs). The key objectives of load balancing and resource allocation are effective resource utilisation, performance optimisation, and cost optimisation. By achieving these objectives, cloud systems may provide effective services to meet user requirements and utilise resources as efficiently and economically as possible. In this study, we propose a novel approach for optimal resource allocation in cloud computing using a hybrid ant colony optimisation and differential evolution (ACO-DE) algorithm. In order to balance the load of VMs from cloud service providers (CSPs), the novel ACO-DE algorithm first generates the shortest routes, then allocates jobs to each VM to distribute resources in the most effective way. The exploration and exploitation abilities of ACO are used to produce an optimal solution, which is further refined using the DE operators. To evaluate the efficacy of the proposed algorithm, various experiments are carried out in a simulated environment to fulfil the key objective. The experiment results revealed the superior performance of the proposed algorithm over the ACO and DE algorithms individually.

**Keywords** Job scheduling · Hybridisation · Resource allocation · Load balancing · Virtual machine · ACO-DE algorithm

---

H. B. Sahoo

CAPGS, Biju Patnaik University of Technology, Rourkela, Odisha 769015, India

D. Chandrasekhar Rao ()

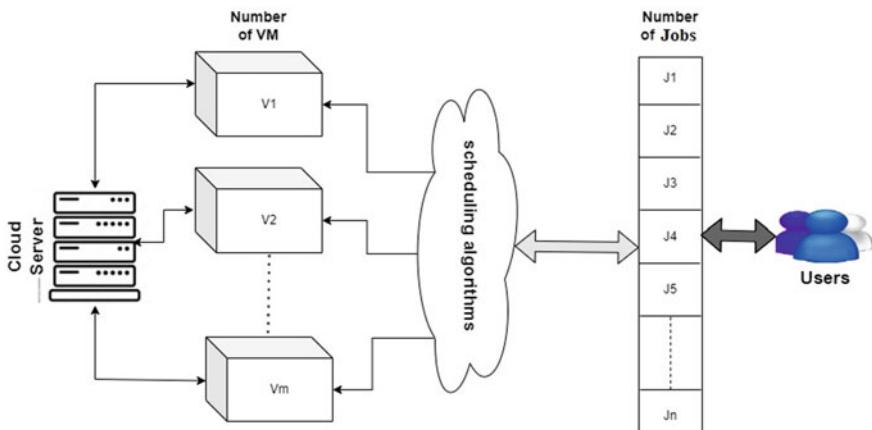
Department of IT, Veer Surendra Sai University of Technology, Burla, Odisha 768018, India

e-mail: [dcrao\\_it@vssut.ac.in](mailto:dcrao_it@vssut.ac.in)

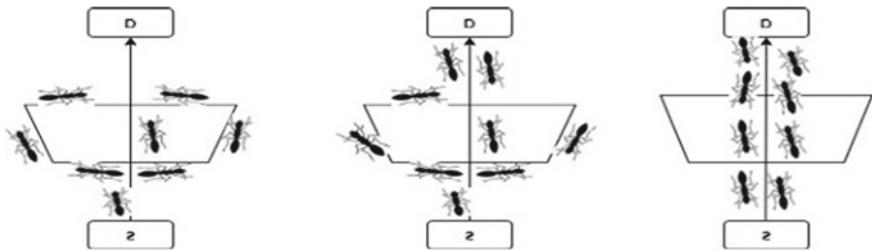
## 1 Introduction

The cloud is a revolutionised version of the computing paradigm in which users can access servers, networks, storage, development tools, and applications online as services [1]. The cloud service providers (CSPs) offer these services by deploying virtual machines (VMs) in response to customer requests for resources [2]. Similar to this, the resource allocator must address the problem of application hunger through effective resource allocation by enabling the service providers to allot the resources for each particular module at a reasonable cost [3]. The influence of infrastructure as a service (IaaS) from cloud providers and load-balanced job scheduling have both grown significantly in recent years. Elastic load balancing is used by Amazon Elastic Compute Cloud (EC2) to distribute client demands. It is important to note that load-balancing job scheduling for cloud computing is an NP-hard problem [4]. Job scheduling tends to involve two layers: the first layer maps user jobs and cloud jobs to the appropriate VMs. Then it specifies the rental period for each VM and distributes the resources to complete the job; the second layer schedules physical machines inside virtual machines and finds the best host to execute the virtual machines in the job queue (see Fig. 1). Numerous challenges must be overcome for cloud computing's resource allocation and load balancing to be efficient and effective [5].

Metaheuristics and heuristic algorithms are complex job scheduling techniques required to address issues with resource heterogeneity, changing workloads, job dependencies, QoS requirements, scalability, energy efficiency, and security [6]. Using cloud-based scheduling, it was chosen for the ant colony optimisation method from a variety of heuristic and metaheuristic approaches. ACO draws influence from animal behaviour in its natural state. Ant colonies are used for social interaction and for finding food sources (see Fig. 2). The pheromone trails that ants leave behind can



**Fig. 1** Scheduling job loads in a cloud computing strategy

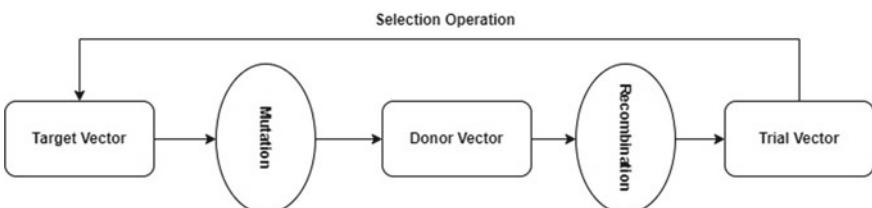


**Fig. 2** ACO principles

be used to reconstruct their travel routes. We may use new job scheduling algorithms based on the results of the various scheduling techniques.

Instead of using more conventional ACO methods, load balancing is done using the improved ACO algorithm [7]. To minimise time and load balance, improve quality, and maximise resource utilisation, we used the proposed hybrid technique to schedule independent jobs on the pool of available resources. After researching numerous genetic algorithms, Storn and Kenneth first put forth the DE algorithm. An initial population of potential solutions, often referred to as vectors, each representing a contender solution to the optimisation issue, forms the basis of the method. New candidate solutions are produced at each iteration of the algorithm by combining and altering current members of the population. Using scaled variations between two randomly selected individuals, a target vector is individually paired to form a trial vector for the mutation process. A selection process is employed to determine if the experimental vectors should replace the corresponding target people in the population [8]. A fitness criterion that assesses each person's worth forms the basis of the selection (see Fig. 3). It is also a group of heuristic search algorithms that are mostly employed to resolve multi-dimensional optimal solutions to space problems. This was done based on the hybrid algorithm, which has a better optimum solution than the other heuristic algorithms like FCFS, RR, etc. We provide the best value in this research based on these objectives by employing hybrid algorithms.

Below is an outline of the remaining parts of the work: the associated literature survey is discussed in Sect. 2, and problem methodologies are provided in Sect. 3. Section 4 includes the proposed algorithm and its detailed explanation. We also include performance and assessment in Sect. 5, which includes the simulation



**Fig. 3** DE principles

of experiments, their recommended procedures, and result analysis. The study is concluded in Sect. 6, which also offers suggestions for additional research.

## 2 Literature Review

The scenarios that follow are drawn from reviews of the literature on load-balancing methods that make use of different algorithms and job scheduling in cloud computing, which will be hybridised, integrated, or improved by numerous authors. To balance the load on communication networks, Schoonderwoerd et al. [9] looked at the idea of mobile software agents inspired by ants. The network supports call as well as a population of basic mobile agents with behaviours based on ants' capacity to build trails. Storn et al. [8] discovered that global optimisation issues over continuous spaces are prevalent in research. The optimisation problem is frequently recognised as a task involving minimisation by the objective function. A practical minimisation technique is typically expected to meet five requirements, including the ability to handle cost functions that are computationally intensive, non-differentiable, nonlinear, and multimodal, as well as ease of use or the requirement for a few control variables to guide the minimisation. Zhang et al. [10] suggested a load-balancing method for an open cloud computing federation based on ant colonies and complex network theory. An open cloud computing federation (OCCF) has numerous applications made up of millions of modules and has dynamic resource and user requirements. The performance of the system is unaffected by the addition of any resource at will, and load balancing can be accomplished quickly in situations with extremely high workloads. The problem is that this approach, which also considers the characteristics of complex networks, vastly enhances the earlier ant colony approaches, which were introduced to provide load balancing in distributed systems. Sharma et al. [7] ensured that resource constraints and activity schedules were taken into account when creating the event-based ACO model to manage dynamic data allocation. Load-balanced ACO (LBACO) is the fastest and most efficient algorithm to use for optimising the objective function. In this research, they will learn how an improved ACO algorithm was used to schedule a variety of tasks to accommodate the complexity of many objectives. Ibrahim and Mahmood [1] discussed several work scheduling strategies used by researchers in the cloud computing environment in their article, which offers information on these strategies. Finally, a large number of authors evaluated the system using a variety of factors, such as cost, throughput, and completion time. Khaleel and Ibrahim [11] suggested the clustered sparrow search algorithm and differential evolution (CSSA-DE), a dual-phase meta-heuristic algorithm. The suggested algorithm's ability to reduce response time while maintaining acceptable energy consumption is made possible by the merging of the sparrow search algorithm and differential evolution. Premkumar et al. [12] argued and concluded that this study investigates and offers innovative variations of the Successive History-Based Adaptive Differential Evolutionary (SHADE) technique

to handle optimal power flow challenges linked to equality and inequality restrictions. As the primary algorithm, SHADE is employed. Investigations have been done into the possibility of an online selection of appropriate methods to aid in parameter tuning as well as the existence of adaptive control factor processes throughout evolution. The static penalty technique is frequently used to eliminate any irrational solutions discovered when looking for workable options. Li et al. [13] introduced an algorithm that optimises the VRP by combining differential evolution (DE) and ant colony optimisation (ACO). The main difference between the ACO and DE is that they may completely leverage each other's strengths to compensate for each other's shortcomings. As a result, this methodology can be improved and applied to different VRPs and other logistics and transportation industries.

In the literature study mentioned above, we discovered two papers: one is a review article based on various methods used in job scheduling in cloud computing, and the other is a paper that will be produced to implement job scheduling in cloud computing. Following this work, we gathered papers that contained the load-balanced ACO method and the differential evolution algorithm. We combined the two algorithms from those studies to create the ACO-DE algorithm, which will be a more efficient and optimal approach overall.

### 3 Methodologies of Problem Statement

We considered the primary obstacle of allocating independent jobs among VMs in a cloud context in order to reduce total make-span time, maintain load balancing, and make the most of available resources, where the computation costs for each job and virtual machine, measured in millions of instructions (MI) and millions of instructions per second (MIPS), are to be kept to a minimum [14]. The first-come, first-served (FCFS) mechanism, which only permits each machine to execute one specific job at a time, has certain assumptions that have been applied to the proposed method in this study. The programme comprises multiple independent jobs that are not prioritised or ranked by computers but whose initial job execution times are approximated or calculated.

#### 3.1 Job Scheduling Methods

Let  $VM_j$  represent a collection of virtual machines that are present in the system, each of which is linked to a  $j$ th job at a time. Each job runs for a specific amount of time, which can be calculated using the jobs and virtual machines computation costs.  $J = \{j_1, j_2, j_3, \dots, j_n\}$  stands for a collection of  $n$  different jobs [14]. The job identification number for each job is  $(j_i)$ , where  $i \in 1 \rightarrow n$ . The definition of the virtual machine is  $VM_j = \{V_1, V_2, V_3, \dots, V_m\}$ . Each machine has a computational cost ( $C_j$ ) and a machine identification number ( $V_{id}$ ), where  $id \in 1 \rightarrow m$ .

### 3.2 Ant Colony Optimisation Methods

The next equation is used to initially set the pheromone value of  $VM_j$ , which is modified at time  $t$ .

$$\tau_j(t) = p_j \times m_j + b_j \quad (1)$$

When the capacity of the communication bandwidth is  $b_j$ , the MIPS of each processor are  $m_j$ , and there are  $p_j$  processors in  $VM_j$ .

$$E_j = ep_j \times em_j + b_j \quad (2)$$

The above equation states that excessive virtual memory ( $E_j$ ) is the computational capacity of  $VM_j$ , which is the sum of all virtual memory accesses made by jobs multiplied by each virtual memory used collectively for all jobs.  $ep_j$  denotes the excessive number of  $VM_j$  processors, and  $em_j$  denotes each processor's MIPS value. The next equation represents the load-balancing factor ( $L_j$ ) employed by the ant approach. A particular  $VM_j$  is more likely to be preferred if its  $L_j$  is higher. The execution time for each virtual machine in the set is then represented by its median value ( $M$ ).

$$L_j = \frac{2M}{EU_j + M} \quad (3)$$

$$EU_j = \frac{\sum_{n=0}^l JL}{E_j} + \frac{F}{b_j} \quad (4)$$

In the equation above,  $EU_j$  stands for the job's expected  $VM_j$  execution time. The length of the job before it is executed determines the size of the input file ( $F$ ). Furthermore, the total job length ( $JL$ ) is the total of all jobs delivered to  $VM_j$ . Here, a probabilistic approach to conventional ACO methods is used to choose a VM for the incoming job, where the  $k$ -ant and  $j$ -job will choose  $VM_j$  for the vacant position [7].

$$p_j^k(t) = \begin{cases} \frac{|\tau_j(t)|^\alpha |E_j|^\beta |L_j|^\gamma}{\sum |\tau_j(t)|^\alpha |E_j|^\beta |L_j|^\gamma}, & \text{if } j \in 1 \dots n \\ 0, & \text{Otherwise} \end{cases} \quad (5)$$

The following equation is used to determine the local pheromone value of  $j$  jobs over  $k$  ants during the upcoming  $t$  time.

$$\tau_k(t+1) = (1 - \sigma) \times \tau_{jk} + \sigma \times \Delta \tau_0 \quad (6)$$

Then the next equation is used to find out the global pheromone value of  $j$  jobs over  $k$  ants during the upcoming  $t$  time [15].

$$\tau_k(t+1) = (1 - \mu) \times \tau_{jk} + \mu \times \Delta\tau_0 \quad (7)$$

Both the local ( $\sigma$ ) and global ( $\mu$ ) pheromone trails have an evaporation rate coefficient of (0, 1]. Let  $T_c$  be the termination condition where the  $j$  number of jobs will terminate, then  $\Delta\tau_j = 1/T_c$ , where  $0 < \Delta < 1$ .

### 3.3 Differential Evolution Methods

By adding randomly selected deviations with a normal distribution to the nominal response, an initial population from the  $N_p$  number of populations might be produced. Then the  $\tau_{k,G}$  is the initial set from the global pheromone value of ACO methods, where  $k = 1, 2, 3 \dots N_p$  of R number of generation [8]. The mutation procedure involves creating a mutant vector by utilising the scaling factor ( $f$ ). The following equation produces a mutant vector for each target vector  $\tau_{k,R}$ .

$$V_{k,R+1} = \tau_{w1,R} + f \cdot (\tau_{w2,R} - \tau_{w3,R}) \quad (8)$$

The random indexes  $w1, w2, w3 \in \{1, 2, 3 \dots N_p\}$  are integers which are mutually different and  $f > 0$ .  $f$  is a real and constant factor  $\in [0,2]$  and subsequently develops the trial vector that may be obtained by employing mutant vectors in the crossover condition, which is done to expand the diversity of the issue. The crossover probability is  $(p) \in 0 \rightarrow 1$ . Then  $\delta$  is the variable location that was chosen at random from 1 to  $k$ , and  $w$  is the random integer  $[0, 1]$ .

$$U_k = \begin{cases} V_k, & \text{if } w \leq p, \text{ OR } k = \delta \\ \tau_k, & \text{if } w > p, \text{ AND } k \neq \delta \end{cases} \quad (9)$$

During the selection process, bounding is necessary in order to choose the offspring by choosing a lower bound value. Following is an analysis of each offspring's fitness function ( $f_U$ ). The impulsive decision is taken once all solutions have generated progeny.

$$\text{if } f_{U_k} < f_k \begin{cases} \tau_k = U_k \\ f_k = f_{U_k} \end{cases} \text{ where } \tau_k \text{ and } f \text{ are the same if } f_{U_j} > f_j. \quad (10)$$

Then finally, it offers the best optimised solution for  $\Delta\tau_k^{\text{best}}$ , which is determined using the below formula, where  $S_{\text{best}}$  denotes the distance travelled by the best ant during its excursion [15]. The finest iteration of  $T_{\text{cmax}}$  may reveal this.

$$\Delta\tau_k^{\text{best}} = \begin{cases} 1/S_{\text{best}} & \text{if } k \in \text{best tour,} \\ 0 & \text{Otherwise} \end{cases} \quad (11)$$

### 3.4 Objective Function

The computation for the first objective function is known as make-span (ms). The second objective function, known as the total cost (TC), is determined by adding the unit time cost ( $\theta_j$ ) and the execution time for each job.  $C(J_n, V_m)$  indicates how long it took to complete the job, where  $st_j$  is the job's start time when the virtual machine is accessible [14].

$$ms = \min(\max(C(J_n, V_m))) \quad (12)$$

$$C(J_n, V_m) = st_j + EU_j \quad (13)$$

$$TC_j = \sum_{n=0}^l JL \times \theta_j \quad TC_j = \sum_{n=0}^l JL \times \theta_j \quad (14)$$

## 4 Proposed Algorithm

**Algorithm 1** Proposed ACO-DE algorithm

---

Input Fitness Function(F), Np, Termination Condition(Tc), Scaling Factor(f), Crossover Probability(Pc)

Output Best Optimized Solution ( $\Delta\tau_k^{\text{best}}$ )

```

1 Begin
2   Initialize Pheromone Trails( $\tau_j$ ),Population set( $P_i$ )
3   While(t = 1 to  $T_c$ ,  $T_c > T_{c\max}$ ,  $T_c = T_c + 1$ ) do
4     For each job j, where j = 1,2,3..... $VM_j$ 
5       Construct a new solution  $\tau_k$  by using the probabilistic rule
6       Update Local pheromone trails
7     End for
8     Update Global pheromone trails
9     For  $\tau_k = 1$  to Np
10      Generate a mutant vector  $V_k$  by using f
11      Generate a trial vector  $U_k$  according to Pc
12    End for
13    For  $\tau_k = 1$  to Np
14      Bound  $U_k$ 
15      Evaluate the fitness( $F_{U_k}$ ) of  $U_k$ 
16      Perform Selection operation  $F_{U_k}$  and  $F_k$  to update  $P_k$ 
17    End for
18  End while
19  Best optimize solution  $\Delta\tau_k^{\text{best}}$  in the population set
20 End

```

---

#### **4.1 Procedure for ACO-DE Hybridisation Algorithm**

In this hybridised ACO-DE algorithm, we do first the initialisation of both the ACO and the DE algorithms, respectively. Then it will go for the first iteration state to meet the proper termination condition ( $T_c$ ) where  $k^{\text{th}}$ -ants are with  $j^{\text{th}}$ -job at time  $t$  for corresponding VMs. Then construct the new solution by using probabilistic rule according to Eq. 5. After arriving at the solution utilising the updated local and global pheromone trails, we carry out the DE algorithm flow by designating the suitable VMs using the constant  $\tau_k$  value for the target vector. According to Fig. 2 and Eq. 8, the mutation happens for generating donor vector or mutant vector ( $V_k$ ) by using scaling factor(f). Then, according to Eq. 9, generate the trial vector ( $U_k$ ) by using crossover probability (Pc) and the mutant vector to get the final solution in the iteration. To perform the selection operation, the corresponding bounding function is chosen for getting the minimum value of bound ( $U_k$ ). Using the evaluation of fitness function ( $F_{U_k}$ ) of  $U_k$ , go for the selection procedure according to Eq. 10. By selecting the corresponding  $\tau_k$  from the current iteration of the selection procedure, update the global pheromone solution. Similar process is executed in every iteration until the termination criteria is satisfied. Then finally find out the best solution by for corresponding job to its VMs from Eq. 11. Finding the best VM for the job to allocate the resources is made easier by using the suggested algorithm to satisfy the related objective functions.

### **5 Result and Analysis**

MATLAB R2021a has been employed for the simulation and comparison of the proposed algorithm with other algorithms in this paper. With CloudSim software, we are going to implement further job scheduling in cloud computing.

#### **5.1 Simulation of Experiments**

We used a Windows 11 processor with an Intel Core i5-8250U CPU clocked at 1.60 GHz and an 8 GB RAM system to obtain our proposed algorithm findings. Utilising the CloudSim Toolkit, the ACO is simulated, and the load balancing and make-span of the performance are assessed. No job is dependent on any other job for the experiment because it only takes into account independent jobs. The link transmission rates are assumed to be evenly distributed between 40 and 10,000 Mbps. The following points provide specifics on the settings used for the simulation analysis, where it shows the value of all the parameters that will be used in the simulation analysis, i.e.  $\alpha = 1$ ,  $\beta = 5$ ,  $\mu = 0.2$ ,  $\rho = 0.5$ ,  $\tau = 0.5$ , Job = 20 → 200, VM = 1 → 10, Ants = 10,  $f = 0 \rightarrow 2$ ,  $Pc = 0 \rightarrow 1$ , and  $\delta = 3$ . Then for the CloudSim Toolkit

parameter, there are three types of values taken: entity, parameter, and value. From the job as Cloudlet, the job length is between 1000 and 50,000, and the total number of jobs is between 10 and 200. From the VM, the total number of VMs is in between 1 and 20, the MIPs rate is between 500 and 2000, and the RAM contains 512 for this operation. From the datacentre, the number of datacentres is 10 and the number of hosts is in between 2 and 6. By testing those three algorithms, ACO, DE, and hybridised ACO-DE, we were able to get these values from the simulation results and input them into Tables 1, 2, 3, and 4, which are listed below.

In this experimental result, we found the below graphs, which provide more information about how the objective of our research work will be more efficient. Tables 1, 2, and 3 show that it is clear that the hybridised ACO-DE algorithm consumes resources at a slightly lower rate than the traditional ACO and DE algorithms overall while using resources at a slightly higher rate overall.

It shows that there is a small time-dependent improvement in the traditional ACO and DE algorithms for the cloud computing job scheduling process compared to the hybridised ACO-DE algorithm. The number of jobs in the process is shown on the graph's x-axis, and the process execution time (see Fig. 4), cost (see Fig. 5), and resource use (see Fig. 6) are shown on the y-axis, accordingly.

**Table 1** Processing result of differential evolution algorithm for different number of jobs

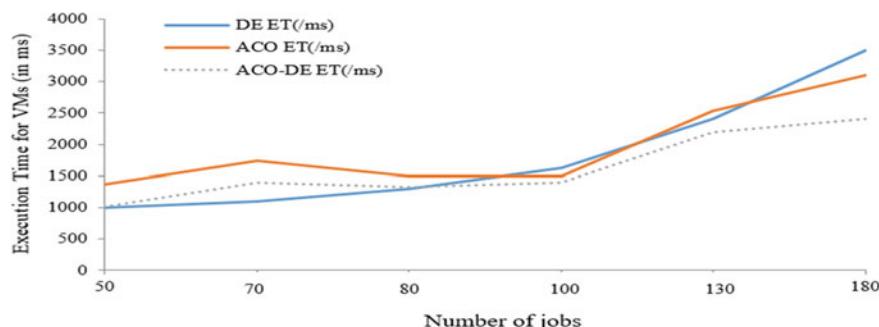
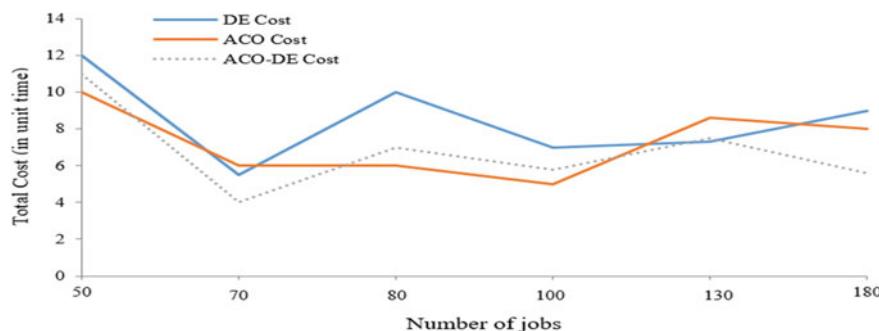
Virtual machine numbers	Virtual machine ID	Number of jobs	Resource consumption rate (%)	Execution time (/ms)	Resource utilisation (%)	Execution cost per unit time
1	2	50	21.52	1000	76.18	12
2	4	70	23.62	1100	79.2	5.5
3	6	80	24.73	1300	74.3	10
4	8	100	25.32	1620	76.48	7
5	9	130	22.35	2400	77.65	7.3
6	5	180	34.56	3500	65.44	9

**Table 2** Processing result of ant colony optimisation algorithm for different number of jobs

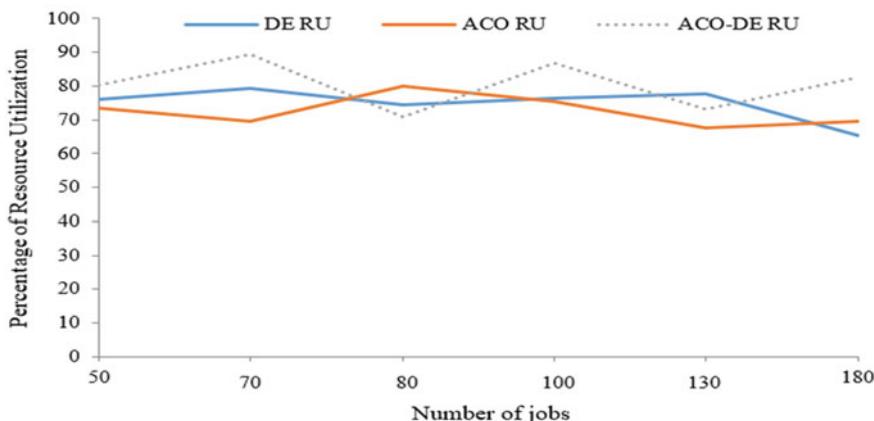
Virtual machine numbers	Virtual machine ID	Number of jobs	Resource consumption rate (%)	Execution time (/ms)	Resource utilisation (%)	Execution cost per unit time
1	2	50	25.46	1370	73.3	10
2	4	70	25.86	1740	69.7	6
3	6	80	26	1490	80	6
4	8	100	24.74	1500	75.26	5
5	9	130	36.81	2530	67.75	8.6
6	5	180	35.46	3100	69.54	8

**Table 3** Processing result of hybridised ACO-DE algorithm for different number of jobs

Virtual machine numbers	Virtual machine ID	Number of jobs	Resource consumption rate (%)	Execution time (/ms)	Resource utilisation (%)	Execution cost per unit time
1	2	50	10.76	1020	80.24	11
2	4	70	12.27	1400	89.4	4
3	6	80	13.12	1320	70.72	7
4	8	100	18.24	1400	86.76	5.8
5	9	130	20	2190	72.99	7.5
6	5	180	17.46	2400	82.54	5.6

**Fig. 4** Make-span (minimum of execution time)**Fig. 5** Cost of job execution

Therefore, a good and efficient scheduling method for cloud job scheduling is to combine the load balancing of the ant colony algorithm with the differential evolution algorithm. It demonstrates how job scheduling with hybridisation is cost-effective for both the user and the cloud service provider, which requires virtual



**Fig. 6** Resource utilisation

machine resources. It also demonstrates how much more of the resources of the virtual computer were consumed.

## 6 Conclusion

This study of several algorithms focuses processing time on the optimisation of cloud computing resource scheduling. Before applying non-dominance sorting to discover the parsimonious group of solutions that represent the conflict between make-span and cloud load balancing, the method employs the ACO approach to find local optimal solutions. By doing so, you increase resource utilisation while satisfying user demands for execution performance. In comparison with previous heuristics, the hybridised ACO-DE algorithm can produce a better solution. By enlarging the system's search space, the DE method significantly improves the ACO algorithm's optimisation performance. This proposed methodology can be improved and applied to a variety of services and other industries. Our future work will aim to significantly decrease computation time and enhance the parallelisation performance of the suggested technique. It may consider optimising multiple objectives at once, including reducing energy consumption, improving QoS, reducing response times, and improving fault tolerance.

## References

- Ibrahim IM (2021) Task scheduling algorithms in cloud computing: a review. *Turkish J Comput Math Educ (TURCOMAT)* 12(4):1041–1053

2. Kinger K, Singh A, Kumar Panda S (2022) Priority-aware resource allocation algorithm for cloud computing. In: Proceedings of the 2022 fourteenth international conference on contemporary computing, pp 168–174
3. Abid A, Manzoor MF, Farooq MS, Farooq U, Hussain M (2020) Challenges and issues of resource allocation techniques in cloud computing. *KSII Trans Internet and Inf Syst* 14(7)
4. Panda SK, Jana PK (2019) Load balanced task scheduling for cloud computing: a probabilistic approach. *Knowl Inf Syst* 61(3):1607–1631
5. Potluri S, Subba Rao K (2020) Optimization model for QoS based task scheduling in cloud computing environment. *Ind J Electr Eng Comput Sci* 18(2):1081–1088
6. Fu X, Hu Y, Sun Y (2020) Cloud computing task scheduling based on improved differential evolution algorithm. In: Proceedings of the 2nd international conference on artificial intelligence and advanced manufacture, pp 118–124
7. Sharma N, Garg P (2022) Ant colony based optimization model for QoS-Based task scheduling in cloud computing environment. *Measurement: Sens* 24(2022):100531
8. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Global Optim* 11:341–359
9. Schoonderwoerd R, Holland O (1999) Minimal agents for communications network routing: the social insect paradigm. In: Software agents for future communication systems. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 305–325
10. Zhang Z, Zhang X (2010) A load balancing mechanism based on ant colony and complex network theory in open cloud computing federation. In: 2010 The 2nd international conference on industrial mechatronics and automation, vol 2. IEEE, pp 240–243
11. Khaleel MI (2023) Efficient job scheduling paradigm based on hybrid sparrow search algorithm and differential evolution optimization for heterogeneous cloud computing platforms. *Internet Things* 22:100697
12. Premkumar M, Kumar C, Dharma Raj T, Jebaseelan SDTS, Jangir P, Alhelou HH (2023) A reliable optimization framework using ensembled successive history adaptive differential evolutionary algorithm for optimal power flow problems. *IET Generat Transm Distrib* 17(6):1333–1357
13. Li H, Zhang X, Fu S, Hu Y (2021) A hybrid algorithm based on ant colony optimization and differential evolution for vehicle routing problem. *Eng Lett* 29(3)
14. Gupta A, Garg R (2017) Load balancing based task scheduling with ACO in cloud computing. In: 2017 International conference on computer and applications (ICCA). IEEE, pp 174–179
15. Dorigo M, Birattari M, Stutzle T (2006) Ant colony optimization. *IEEE Comput Intell Mag* 1(4):28–39

# Prediction of Cardiovascular Disease by Feature Selection and Machine Learning Techniques



Aditya Ranade and Nitin Pise

**Abstract** Cardiovascular diseases (CVDs) are prevalent in the population and often lead to fatalities. Recent polls indicate that the death rate is increasing due to people's increased use of tobacco, high blood pressure, cholesterol, and obesity. These factors also exacerbate the severity of the disease. Therefore, it is crucial to conduct research on the variations of these factors and their impact on CVD. To prevent further disease progression and reduce mortality rates, it is essential to utilize current procedures. Various techniques, such as AI and data mining, are available to predict CVD precursors and detect their behavior patterns in large amounts of data. The results of these forecasts will assist clinical experts in decision-making and early diagnosis, reducing the likelihood of patient fatalities. This study investigates and comments on the Exhaustive Feature Selection (EFS) and Sequential Feature Selection (SFS) techniques and the results obtained using them with various classifiers. The paper also provides an overview of the current methods based on features and algorithms used.

**Keywords** Machine learning · Cardiovascular disease · Exhaustive feature selection · Sequential feature selection · Bagging method

## 1 Introduction

The term cardiovascular disease (CVD) refers to a group of ailments that affect either the heart or the arteries of the human body. It can also result in damage to blood vessels in organs such as the kidneys, heart, eyes, and brain. CVD is responsible for a significant number of deaths among young people in many industrialized nations and some developing countries.

---

A. Ranade (✉) · N. Pise

Dr. Vishwanath Karad, MIT World Peace University, Pune 411038, India  
e-mail: [aranade1@gmail.com](mailto:aranade1@gmail.com)

N. Pise  
e-mail: [nitin.pise@mitwpu.edu.in](mailto:nitin.pise@mitwpu.edu.in)

CVDs can be classified into four main groups. The primary condition is coronary heart disease, which occurs when blood flow to the heart muscle is obstructed. This can lead to angina, heart attacks, and heart failure. However, accurate diagnosis and treatment of cardiovascular disease can be challenging due to the lack of diagnostic tools, healthcare workers, and other resources.

The complexity of early-stage testing procedures for the diagnosis of heart disease has been found to be one of the main factors affecting the quality of life of cardiovascular patients. This is especially true in developing nations. However, recent studies have shown that machine learning systems can effectively evaluate medical data and identify diseases.

The goal of this publication is to explore feature selection techniques in machine learning for predicting cardiac disease. The paper will examine various machine learning algorithms and feature selection techniques to determine which methods are the most effective and advantageous.

## 2 Literature Review

Mohan et al. [1] have proposed a hybrid technique. The Hybrid Random Forest and Linear Method (HRFLM) algorithm was implemented to identify factors for cardiovascular disease in the UCI Cleveland database. Three aggregation rules—Apriori, Predictive, and Tertius—were applied. The HRFLM algorithm makes use of an ANN with back propagation, along with 13 clinical features as input. Results obtained were then compared to those from traditional methods such as Random Forest, Decision Tree, and K-Nearest Neighbors (KNNs). It was observed that HRFLM had better accuracy when compared to these traditional methods.

Ganesan et al. [2] have used four different algorithms—J48, Support Vector Machine, Logistic Regression, Multi-layer Perceptron. The proposed model incorporates three inputs (IoT Sensors, a Heart disease dataset, and Patient records) to predict whether a person will have heart disease. The Cleveland dataset from the UCI machine learning archive (270 instances and 13 characteristics) was utilized in their analysis. After preprocessing the dataset and running the trials, the J48 algorithm surpassed SVM and MLP with 91.48 percent accuracy, whereas MLP and SVM achieved 78.14 percent and 84.07 percent accuracy, respectively.

Authors in [3] have implemented three machine learning techniques: Decision Tree, Random Forest, and Hybrid Model. The Hybrid Model is developed using Decision Tree and Random Forest techniques. The training data and the random forest probabilities are combined and fed into the decision tree algorithm. The probability of a decision tree being derived from test data is likewise linked and presented in an analogous manner. There are 270 types of circumstances and 14 attributes in the Heart Statlog dataset. In order to remove irrelevant and missing data, the Dataset is processed before it is loaded. After data cleaning, particle swarm optimization (PSO) is applied on the dataset. Six of the weakest points were removed, with seven strong

points chosen from that list. The amalgamated model has the highest accuracy of 88.5.

In [4], Kigka et al. have examined four different machine learning algorithms;

1. Artificial Neural Network,
2. Support Vector Machine,
3. Random Forest,
4. J48.

The Gain Ratio Algorithm, Principal Component Analysis, and Attribute Evaluation Technique were used to implement feature selection. By properly customizing the input characteristics and using a forward selection Procedure, the sensitivity and specificity ratio may be improved over time.

Singh et al. [5] used fuzzy model for prediction. A new approach to the analysis of community health survey data that includes both SEM and FCM may yield useful results that can help prevent cardiovascular mortality. The relationship between CCC 121 and 20 components was used to develop a structural equation model (SEM). These SEM models were fitted with the training data, with standardized parameter estimates as edges. The left nodes specified by the drivers =, or, and the variables are represented by their corresponding values (-1, 1). The right node uses the “=” driver for passive variable expression, “” driver for regression, and “” driver for covariance. The edges are directed from the right-hand side (RHS) to the left-hand side (LHS) node. The driver is bidirectional regardless of “=” or “”. These edges from the SEM chart were used to create the  $7 \times 27$  weight matrix in our FCM chart. The 20 items from the CCHS data set we mentioned are represented by bumps from 1 to 20 on the weighting matrix. Node 21 indicates the true value of CVD, CCC 121, and bumps from 22 to 27 indicate our six passive variables (GH, CC, RA, PA, INC and RCVD).

Rajjliwal and Chetty [6] have used deep learning techniques. LASSO Regression has been used for feature selection. The super dataset was created by combining the NHANES subsets from 1999–2000 to 2015–2016. The combined super dataset contains information on about 37,079 individuals, including their CVD complaint status, as well as information from their demographic, examination, laboratory, and questionnaire data. With 1300 individuals having a positive CVD status and 35,779 cases having a negative CVD status, the super dataset was significantly skewed (no heart complaint). adverse CVD (no heart disease).

In [7] Chowdhury et al. used a questionnaire. The questionnaire's development was assisted by Dr. Muhammad Shahabuddin, Head of the Cardiology Department at Sylhet MAG Osmani Medical College. He proposed eighteen qualities, highlighting the eight primary qualities of them. Here are some of them: 1.age, 2.gender, 3. Use of tobacco, 4. Diabetology 5. Hypercholesterolemia in adults, 6.Chest pain, 7. Family History, 8. Hypertension (140/90). On experimentation SVM has the highest accuracy with all attributes and 8 most relevant attributes.

Lakshmanarao et al. [8] have proposed a model wherein they used ANOVA and Mutual Information for feature selection. The experimentation has been conducted on two datasets. The UCI machine learning repository provided dataset 1.14 features are

present in 303 samples in Dataset 1. The remaining features (age, gender, tptrestbps, chol, fbs, etc.) are independent features with the exception of the “target” feature. A total of 4238 samples with 16 features are in Dataset-2. The remaining features (male, age, education, current smoker, cigsPerDay, BPMeds, bmi, etc.) are regarded as independent features, while the “TenYearCHD” feature is a dependent feature. To test different classifiers, they have used a variety of them. The voting classifier’s accuracy for dataset 1 was 90%. For dataset 2, accuracy of 99% was attained with stacking classifiers in conjunction with random oversampling technique after using all three sampling techniques with various classifiers. SMOTE also provided a good accuracy of 93%, even though random oversampling had a 99% accuracy rate.

Mistry and Wang [9] have employed a hybrid method (HRFLM) to predict Heart Disease. The initial step of the machine learning process begins with the pre-existing data, which is then used to select features based on Decision Tree probability, constructing models which measure performance and classification and ultimately leading to an elevated accuracy. As variables and criteria for collecting datasets, Decision Tree characterizations were deployed. Afterward, classifiers were used to appraise the accuracy of each collective dataset. The finest models from the available data are chosen based on their minimal error rate. By selecting a Decision Tree cluster with such a low error rate and gathering the corresponding classifier features, we can further optimize performance. The dataset applied was obtained from the UCI machine learning repository.

Using a dataset from the Kaggle database, Jinjri et al. [10] analyzed and compared algorithms selected for this study. Three input features are present in the dataset, which has a collection of 77,000 clinical trial records including data collected from hospitals regarding cardiovascular-related diseases. The dataset also contains 11 attributes, including 1 target variable with the label “(Absence or Presence) for diagnosis,” 4 examination features, 4 subjective features, and 3 objective features. Five classification algorithms—DT, LR, KNN, NB, and SVM—have been tested. The most accurate classification algorithms are SVM and logistic regression, with 72.66% and 72.33%, respectively.

Xu et al. [11] have proposed a novel imputer method to handle missing values. First, a clinician with advanced education incorporates cardiovascular information on important clinical variables. A case may contain multiple records of various medical issues. If any value in a particular variable in an event record is replaced with “NA”, a value is missing (missing). All records related to the event get the same event ID Second, we prioritized processed patient data matrix.  $Mn \times m$  To create a matrix of, where n is the number of records and m is the number of variables, we are going to use zn Score standardization on numeric variables as a hot representation of categorical variables Let’s build a matrix no Third, two independent encoders, encoder1 and encoder2, which independently practice embedding in record and case conditions, receive information from preprocessed case data  $Mn \times m$ . The program can learn how to project patient data from scratch the area moves to an inactive area, as a result, the case record displays the bed instead. Also, Encoder 2 has retracted levels that provide activation functions by means of a preventable direct unit. Fourth, this system this calculates error errors only for observed values using input and imputed data

Specifically, for numeric values and the cross entropy error with respect to range values, we calculate the squared error.

### 3 Feature Selection Techniques

Feature selection techniques are an important part of preprocessing as irrelevant features may hamper the results. The most relevant features need to be extracted for the best and accurate results. Selection of features also helps to reduce the execution time of the classifier and optimizes space and time complexity of the algorithm. The feature selection techniques used in this study are: Exhaustive Feature Selection Technique and Sequential Feature Selection Technique.

#### 3.1 EFS Feature Selection Technique

In exhaustive feature selection, every possible combination of the dataset's features is tested against the performance of a machine learning algorithm. The feature subset with the best results is chosen. The pseudo-code for the algorithm is as follows:

Input: Set of features  $X$ , size of feature set  $n$ , size of target feature subset  $d$ , set of possible feature subsets,  $F$ , of  $X$  where each subset is of size  $d$  Output: Optimum feature subset  $Y_{opt}$  of size  $d$ .

##### Algorithm 1 EFS Algorithm

1.  $Y_{opt} \leftarrow \emptyset$
2.  $G_{opt} \leftarrow -\infty$
3. for all  $Y_i \in F$  =  $\{Y_0, Y_1, \dots, Y_k\} \mid k = (do$
4.    $G_i \leftarrow J(Y_i)$
5.   if  $G_i > G_{opt}$  then
6.      $Y_{opt} \leftarrow Y_i$
7.      $G_{opt} \leftarrow G_i$
8.   end if
9. end for

#### 3.2 SFS Feature Selection Technique

Algorithm: Sequential Forward Selection.

##### Algorithm 2 SFS Algorithm

1. Start with the empty set  $Y_0 = \{0\}$

2. Select the best next feature  $x^* = \arg \max = [\sqrt{(x+x)}]$
3. Update  $Y + 1 = Y_2 + x + ;k = k + 1$
4. Go to 2

In essence, a subset of wrapper methods that sequentially add and remove features from the data are sequential feature selection algorithms. This method is known as “spontaneous selection of features” when this happens. When each feature is evaluated separately, it selects  $M$  features from  $N$  features on the basis of individual scores. Because it doesn’t take feature dependence into account, it only occasionally works. The pseudo-code for the algorithm is as follows.

## 4 Performance Measure Indices

Key performance indicators such as accuracy, precision, recall and F1 score can be used to assess the efficiency of a model. The factors deciding the above indicators are as follows:

The formulas for the same are as follows

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (1)$$

$$\text{Precision} = \frac{(TP)}{(TP + FP)} \quad (2)$$

$$\text{Recall} = \frac{(TP)}{(TP + FN)} \quad (3)$$

$$\text{F1-score} = \frac{2(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (4)$$

## 5 Proposed Approach for Classification

### 5.1 Dataset Details

Various different datasets have been used for this study. They are as follows

1. Framingham Dataset [12]
2. UCI Machine Learning Repository [13]

The Framingham dataset has 16 features and 4028 records. The features are as shown in Table 1.

**Table 1** Attributes of Framingham dataset

No.	Attribute	Data category	Details
1	Age	Continuous	Patient's age
2	Sex	Nominal	Patient gender
3	Education	No info	Education of the patient
4	Current Smoker	Nominal	The smoking status of the patient
5	BP meds	Nominal	Whether or not the patient was on blood pressure medication
6	Cigs per day	Continuous	The average daily cigarette consumption of an individual
7	Prevalent Hyp	Nominal	The presence or absence of hypertension in the patient
8	Prevalent stroke	Nominal	The occurrence of a stroke in the patient's medical history
9	Diabetes	Nominal	The occurrence of diabetes in the patient's medical history
10	Sys BP	Continuous	Systolic blood pressure
11	Tot Chol	Continuous	The overall level of cholesterol in the body
12	Dia BP	Continuous	Diastolic blood pressure
13	Glucose	Continuous	Glucose level
14	Heart rate	Continuous	Heart rate
15	CHD	Binary	10 year prediction

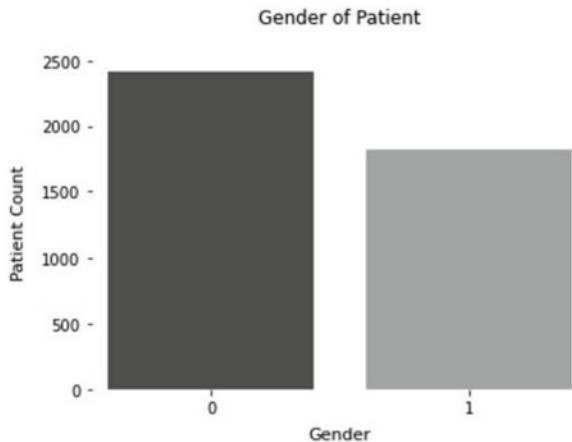
Figure 1 shows the gender dissemination in the dataset. There are around 2500 males and 1500 females in the dataset where “1” represents Male and “0” represents a Female. The UCI Machine Learning Repository has 16 features and 920 records. In this section, we will explore the different machine learning methods employed in developing a cardiovascular prediction system. A brief description about the various algorithms has been discussed in this section.

## 5.2 KNN (*K*-Nearest Neighbors) [9]

This nonparametric algorithm uses proximity to classify and predict a single data point’s grouping with the  $k$ -nearest neighbor algorithm. While it is possible to apply this algorithm to classification or regression, it is usually used as a grading algorithm that assumes that similar points must be located nearby. It calculates the distance from two points with three different methods, namely

1. Euclidean distance
2. Manhattan distance
3. Minkowski distance.

**Fig. 1** Gender dissemination for Framingham dataset



### 5.3 Logistic Regression [15]

For forecasting analysis and categorization this type of statistics model, which is called the logit model, has frequently been applied. The probability of the event occurring, such as whether there is a vote or no vote, shall be calculated using statistical regressions based on an aggregated set of independently variable data. Since the result is uncertain, the range of the dependent variable is from 0 to 1.

### 5.4 Decision Tree [16]

In summary, a Decision Tree can be described as a method of supervised learning, which is used to classify and to perform regression. Its structure can be described as having a root node, branches, internal and leaf nodes. It is a type of divide and conquer approach that uses a greedy search technique to find the ideal split points within a tree by using a greedy search algorithm. Upon categorizing a large number of records under distinct labels for distinct classes, this separation procedure is then repeated in a top-down, recursive manner in order to separate all records into different categories.

### 5.5 Support Vector Machine (SVM) [14]

Support Vector Machine, or SVM, which is one of the most efficient algorithms when it comes to supervised learning. SVM is used for classification and regression applications as well. Through the mapping of data to a large-dimensional feature space, SVM categorizes points regardless of their linear differentiation. When a

separation between categories has been determined, the data is converted to give a hyperplane representation.

## 5.6 *Bagging Method*

Bagging is an aggregate approach that aggregates the forecasts of all models, when they are linked to different subsets in a training dataset. With bagging, random sampling and replacing data are carried out on an individual training set so that a series values can be selected by different methods. These weak models are individually trained and dependent on the task following the collection of several data samples. Four ensembles of hybrid models, on the basis of DT, LR, SVM, and KNN, were developed using Bagging techniques.

## 6 Implementation

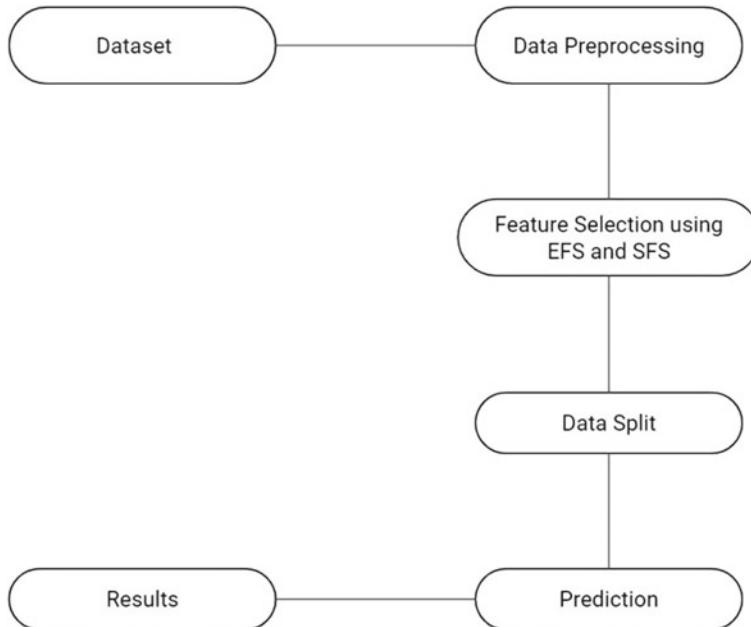
There are five datasets that have been used in this study: The Cleveland, Hungary, Switzerland, VA Long Beach and Framingham dataset. The first four have been amalgamated into one dataset for easier and faster execution. Details of the implementation process are described in the sections below. Figure 2 shows the proposed methodology for the experiment.

### 6.1 *Different Machine Learning Libraries and Feature Selection Techniques Used*

The EFS and SFS technique has been used to select the features. Feature selection is an important part of data preprocessing as not all features of the dataset are relevant to the problem trying to be solved hence selecting of relevant features is a must. Eight classifiers have been used in this study:

1. Logistic Regression
2. Support Vector Machine
3. K-Nearest Neighbor
4. Decision Tree Classifier
5. Logistic Regression Bagging
6. Support Vector Machine Bagging
7. K-Nearest Neighbor Bagging
8. Decision Tree Classifier Bagging

Two techniques have been used to remove the imbalance within the datasets:  
Near Miss method



**Fig. 2** Proposed methodology

SMOTE [17]

## 6.2 A Summary of Data Preprocessing and Cleaning Techniques

In the contemporary era, a lot of data can be obtained through surveys, experiments, the internet, and other means. However, the data will undoubtedly have absent values, noise, and anomalies. The datasets used in this sample used for this study also include null or missing values. In the Framingham dataset, the null values have been dropped. The education column which is irrelevant to predicting whether a person will have cardiovascular disease has also been dropped from the dataframe. In the UCI machine learning dataset the null values have been replaced with the mean of the attribute. Data must also be normalized or standardized before machine learning techniques can be used.

## 7 Results

### 7.1 Outcomes of Feature Processes Selection

This part contrasts the results of various classification models using various input characteristics. Four machine learning models and four hybrid methods were used after some pertinent features had been extracted using the Exhaustive Feature Selection Technique and Sequential Feature Selection Technique. Finally, the classifiers and hybrid techniques were compared for analysis to predict the better feature selection technique. Additionally, various success measures are examined in order to assess the anticipated results. The suggested models were assessed using the performance measures Accuracy, Precision, F1 score, and Recall.

### 7.2 A Comparison Based on Accuracy of Different Methods

In general, accuracy is considered as the most important way to assess machine learning methods. With respect to four features, the DTC Bagging Classifier turned out to be the most accurate forecast with an accuracy of 70.982%. This accuracy is achieved by the classifier when the data is undersampled using the NearMiss algorithm. The DTC normal classifier produces the lowest accuracy (35.8%). With the 4 EFS features, we achieve accuracy of 57.1, 62.6, 35.8, and 42.1% for the LR, SVM, DTC, and KNN classifications, respectively. LR gives the highest accuracy (84.4%) when the SFS algorithm is used for feature selection. DTC Classifier has the lowest accuracy of 37.2. The hybrid Bagging Classifiers have an accuracy range of 73–79%. There is a significant bump for the hybrid model accuracies. With EFS, the accuracies were in the range of 69–70%.

### 7.3 A Comparison Based on Precision of Different Methods

The LR Bagging model had a notable precision score of 76% when using EFS technique for feature selection. DTC got the lowest precision score at 16.4%. When the data was oversampled with the EFS method the SVM Bagging Classifier achieved the highest precision of 81% and DTC the lowest with 22%. Precision scores for SFS technique with undersampling lie in the range of 16–78%. Majority of the classifiers are closer to 70%. Even when the same data was oversampled with SFS technique, the LR Bagging technique gives the highest precision of 78% and DTC the lowest.

**Table 2** Results for the Framingham dataset when undersampled using SFS (60/40)

Classifier	Accuracy	Recall	Precision	F-score
LR	84.39	2.10	62.50	4.10
SVM	59.40	55.50	20.50	30.00
DTC	37.20	75.00	16.70	27.30
KNN	54.40	58.09	18.90	28.59
LR bagging	74.40	52.00	78.00	58.00
SVM bagging	73.80	60.00	77.00	65.00
DTC bagging	79.16	37.00	76.00	42.00
KNN bagging	77.82	51.00	77.00	57.00

#### 7.4 A Comparison Based on Recall of Different Methods

An essential performance matrix is recall or sensitivity score since it's crucial that persons with heart disease are correctly identified. In the Framingham dataset when the EFS method is used with the data being undersampled the DTC classifier has the highest recall of 75%. LR Bagging and SVM Bagging are tied at 57%. The DTC Bagging classifier has the least recall of 33%. When the data is oversampled, the DTC Bagging classifier has the highest recall of 69%. The DTC classifier has the lowest recall of 43.2%. When the SFS technique is used with undersampling; the DTC classifier has the highest recall of 57%. LR classifier has the lowest recall of 2.1%. SVM, KNN, DTC Bagging, SVM Bagging, KNN Bagging have a recall score of 55.5%, 58.1%, 60%, 37%, and 52% respectively. Table 2 shows the results for the Framingham dataset when undersampling is done with SFS.

#### 7.5 A Comparison Based on F1-Score of Different Methods

The F1-score assesses the effectiveness of different classifiers on diverse datasets by calculating the harmonic mean of accuracy and recall scores. In the first dataset, the LR Bagging classifier achieved the highest F1-score of 63% using the EFS technique and undersampled data. The KNN classifier had the lowest F1-score of 24.2%. When the data was oversampled, the DTC Bagging classifier achieved the highest F1-score of 72%, while the KNN classifier had the lowest F1-score. The SVM Bagging and DTC Bagging classifiers achieved the best F1-scores of 74% on the same dataset when oversampling was used. In the UCI dataset, the DTC Bagging classifier achieved the highest F1-score of 46% with EFS and oversampling, while the KNN classifier had the lowest F1-score of 27.5%. Table 3 shows the results for the Framingham dataset when undersampling is done with EFS.

**Table 3** Results for the Framingham dataset when undersampled using EFS technique

Classifier	Accuracy	Recall	Precision	F-score
LR	57.09	51.30	18.60	27.30
SVM	62.60	42.40	19.00	26.30
DTC	35.80	75.00	16.40	26.90
KNN	42.10	58.90	15.30	24.20
LR bagging	70.68	57.00	76.00	63.00
SVM bagging	66.81	57.00	75.00	63.00
DTC bagging	70.98	33.00	73.00	37.00
KNN bagging	69.79	41.00	73.00	47.00

## 7.6 Accuracy Comparison Table of Proposed Models

Shown in Table 4 is the comparison of the various algorithms. The table gives a brief understanding about the results of experimentation. After conducting an experiment on specific characteristics, the LR model demonstrated the highest accuracy rate (84.4%), while the KNN model yielded a lower accuracy score (61.6%). In our study.

As a general rule, the debate demonstrated that different classification schemes were adequate when compared to previous research, but there are some drawbacks such as relying on particular feature selection methods like an increased emphasis on EFS and SFR for very accurate results. Having a large amount of missing values in the dataset may also be detrimental. If there is a sufficiently important missing value, we have demonstrated how the problem can be solved by proper techniques and therefore further datasets used in our model need to deal with it. Given that the training data is quite large, this model would have been much more precise compared to a larger

**Table 4** Comparison of accuracy of different models for EFS And SFS techniques

Classifier	Accuracy (In percentage)					
	60/40		70/30		80/20	
	EFS	SFS	EFS	SFS	EFS	SFS
LR	64.20	84.40	63.00	83.80	65.40	85.50
DTC	67.00	68.50	70.50	68.50	72.00	71.50
SVM	63.40	66.70	64.30	65.00	64.30	64.70
KNN	61.60	64.40	63.00	65.00	63.40	64.70
LR bagging	70.00	74.00	65.80	66.30	65.40	66.00
DTC bagging	75.00	84.00	75.30	74.90	75.10	75.00
KNN bagging	75.00	78.00	65.80	63.30	65.10	62.70
SVM bagging	64.00	62.00	63.10	65.40	63.40	64.30

dataset. When you select additional features, the results will also be changed. The accuracy, precision, f-score and recall may improve even more with more features.

## 8 Discussion

It was observed that the Sequential Feature Selection (SFS) technique yielded better results than the Exhaustive Feature Selection (EFS) technique in this experiment. However, the number of features selected by both techniques differed, so it cannot be conclusively stated that SFS is the better technique. EFS uses a greedy approach to search for the most relevant attributes in a dataset, which increases time complexity as the number of features increases. This can sometimes take hours or days to complete a single execution.

The LR method has been observed to give the best results when used with the SFS technique. Oversampling and undersampling the data have also played important roles, with the SMOTE algorithm producing better results than undersampling with the Near Miss algorithm. Both algorithms produced different results in experimentation, and different train/test splits also produced varying results.

The 70–30 train/test split parameters produced more consistent results, with less variance observed in various performance indices using this parameter.

## 9 Conclusion

Cardiovascular diseases worsen and become uncontrollable. Complex heart conditions claim the lives of many people every year. In a short period of time, catastrophic consequences could occur if the patient ignores the typical heart disease symptoms. Sedentary habits and high levels of stress in the modern world have exacerbated the problem. It is possible to control the disease if it is discovered early enough. But daily exercise and quitting bad habits as soon as you can are always necessary. Early diagnosis of heart disease may have a profound effect on long-term mortality regardless of social or cultural background. Accurate prediction of the risk is a fundamental step in pursuing this aim. In some of the research, machine learning has already been employed to predict cardiac disease. A similar approach was adopted by the present work, but it applies a more advanced method and an increased dataset to train this model. This study indicates that a number of machine learning algorithms may be utilized with the EFS and SFS feature selection technique to produce a strongly linked feature collection.

This study oversaw the use of five different datasets and two feature selection techniques. Eight different classifiers were also used for the task of prediction. The methods were compared and analyzed to predict the best model to use for predicting a person's probability of getting cardiovascular disease.

## References

1. Mohan S, Thirumalai C, Srivastava G (2019) Effective heart disease prediction using hybrid machine learning techniques. In: IEEE Access, vol 7, pp 81542–81554. <https://doi.org/10.1109/ACCESS.2019.2923707>.
2. Ganesan M, Sivakumar N (2019) IoT based heart disease prediction and diagnosis model for healthcare using machine learning models. In: 2019 IEEE International conference on system, computation, automation and networking (ICSCAN), Pondicherry, India, pp. 1–5. <https://doi.org/10.1109/ICSCAN.2019.8878850>
3. Shree Raksha GM, Hegde R, Shivani MN, Shrinidhi PS, Thashwin Monnappa, MM, Soumyasri SM (2022) A novel technique for prediction of cardiovascular disease. In: 2022 IEEE international conference on data science and information system (ICDSIS), Hassan, India, pp 1–5 (2022)
4. Kigka VI et al (2018) A machine learning approach for the prediction of the progression of cardiovascular disease based on clinical and non-invasive imaging data. In: 2018 40th annual international conference of the IEEE Engineering in medicine and biology society (EMBC), Honolulu, HI, USA, pp. 6108–6111. <https://doi.org/10.1109/EMBC.2018.8513620>
5. Singh M, Martins LM, Joannis P, Mago VK (2016) Building a cardiovascular disease predictive model using structural equation model and fuzzy cognitive. In: 2016 IEEE international conference on fuzzy systems (FUZZ-IEEE), Vancouver, BC, Canada, pp 1377–1382. <https://doi.org/10.1109/FUZZ-IEEE.2016.7737850>
6. Rajjiliwal NS, Chetty G (2021) Deep learning based decision support framework for cardiovascular disease prediction. In: 2021 IEEE Asia-Pacific conference on computer science and data engineering (CSDE), Brisbane, Australia, pp 1–12. <https://doi.org/10.1109/CSDE53843.2021.9718459>
7. Chowdhury MNR, Ahmed E, Siddik MAD, Zaman AU (2021) Heart disease prognosis using machine learning classification techniques. In: 2021 6th international conference for convergence in technology (I2CT), Maharashtra, India, 2021, pp 1–6 <https://doi.org/10.1109/I2CT51068.2021.9418181>
8. Lakshmanarao A, Srisaila A, Kiran TSR (2021) Heart disease prediction using feature selection and ensemble learning techniques. In: 2021 Third international conference on intelligent communication technologies and virtual mobile networks (ICICV), Tirunelveli, India, pp 994–998. <https://doi.org/10.1109/ICCV50876.2021.9388482>
9. Mistry S, Wang L (2022) Efficient prediction of heart disease using cross machine learning techniques. In: 2022 IEEE Asia-pacific conference on image processing, electronics and computers (IPEC), Dalian, China, pp 1002–1006. <https://doi.org/10.1109/IPEC54454.2022.9777309>
10. Jinjri WM, Keikhosrokiani P, Abdullah NL (2021) Machine learning algorithms for the classification of cardiovascular disease—a comparative study. In: 2021 International conference on information technology (ICIT), Amman, Jordan, 2021, pp 132–138. <https://doi.org/10.1109/ICIT52682.2021.9491677>
11. Xu D, Sheng JQ, Hu PJ-H, Huang T-S, Hsu C-C (2021) A deep learning-based unsupervised method to impute missing values in patient records for improved management of cardiovascular patients. IEEE J Biomed Health Inform 25(6):2260–2272. <https://doi.org/10.1109/JBHI.2020.3033323>
12. Framingham dataset <https://www.kaggle.com/datasets/captainozlem/framingham-chd-preprocessed-data>
13. Janosi A, Steinbrunn W, Pfisterer M, Detrano (1988) UCI machine learning heart disease Robert. Heart Disease. UCI machine learning repository. <https://doi.org/10.24432/C52P4X>
14. Singh P, Kumar Pal G, Gangwar S (2022) Prediction of cardiovascular disease using feature selection techniques. Int J Comput Theo Eng 14(3):97–103
15. Nikam SB, Mhaske A, Mantri S (2020) Cardiovascular disease prediction using machine learning models. In: 2020 IEEE Pune section international conference (PuneCon), Pune, India, pp 22–27. <https://doi.org/10.1109/PuneCon50868.2020.9362367>

16. Krithika DR, Rohini K (2021) Ensemble based prediction of cardiovascular disease using Bigdata analytics. In: 2021 international conference on computing sciences (ICCS), Phagwara, India, pp 42–46. <https://doi.org/10.1109/ICCS54944.2021.00017>
17. Fitriyani NL, Syafrudin M, Alfian G, Rhee J (2020) HDPM: an effective heart disease prediction model for a clinical decision support system. IEEE Access 8:133034–133050. <https://doi.org/10.1109/ACCESS.2020.3010511>

# Performance Evaluation and Comparative Analysis of Machine Learning Techniques to Predict the Chronic Kidney Disease



Majid Bashir Malik , Mohd Ali, Sadiya Bashir, and Shahid Mohammad Ganie 

**Abstract** Chronic kidney disease is one of the most fatal diseases affecting people worldwide. As a result, it is critical to forecast and identify this illness as early as possible, enabling medical professionals and patients to take the necessary and appropriate steps. In this paper, we propose a framework using machine learning techniques including logistic regression (LR), naive Bayes (NB), support vector machine (SVM), and decision tree (DT) for better prediction of chronic kidney disease. A publicly available dataset containing 25 parameters and 400 records of patients was employed. We conducted a thorough exploratory data analysis to improve the quality assessment of dataset. The DT method outperformed the other three algorithms by achieving the highest training and testing accuracy rate as 100% and 99.16%, respectively. To validate the model, other performance evaluation metrics were calculated. The model has provided a better predicted outcome when compared to similar research studies. Our proposed model can be used in the healthcare industry for better decision making regarding chronic kidney disease.

**Keywords** Chronic kidney disease · Exploratory data analysis · Machine learning · Decision tree · Support vector machine · Naïve Bayes · Logistic regression

## 1 Introduction

A disorder known as chronic kidney disease (CKD) damages the kidneys and hinders their normal function [1]. Long-term, sluggish organ/cell degeneration is referred to as a chronic disease. When both kidneys are damaged, a common type of kidney

---

M. B. Malik · M. Ali · S. Bashir

Department of Computer Sciences, Baba Ghulam Shah Badshah University, Rajouri 185234, India

S. M. Ganie 

AI Research Centre, School of Business, Woxsen University, Hyderabad 502345, India

e-mail: [shahid.mohammad@woxsen.edu.in](mailto:shahid.mohammad@woxsen.edu.in)

illness called chronic kidney disease develops, and CKD patients suffer from this condition over an extended period. The kidneys are essential organs that filter waste from the blood, manage electrolyte balance, and regulate blood pressure, among other key tasks [2]. These processes may be hampered by kidney injury, which can result in several issues. Reduced kidney function, an estimated glomerular filtration rate [F]  $60 \text{ ml/min}, 73 \text{ m}^2$ , or signs of kidney damage such as albuminuria, hematuria, or abnormalities shown on imaging that have been present for at least three months are all considered to be symptoms of chronic kidney disease (CKD) [3].

Today, chronic kidney disease is a serious global health concern. About 10% of people worldwide were affected by chronic kidney disease [4]. Patients receiving renal replacement therapy (RRT) around the world predominately reside in Europe, Japan, or North America. Contrarily, less than 10% of Indian patients with end-stage kidney disease (ESKD) undergo RRT, and up to 70% of those who begin dialysis either pass away or discontinue their care owing to expense within the first three months. In India, there are around 150–200 cases of end-stage renal disease (ESRD) per 10 lakhs population, while the prevalence of chronic kidney disease is approximately 800 cases per 10 lakhs population. According to a recent medical report, 324 million people worldwide have CKD [5]. A common CKD screening test is the glomerular filtration rate (GFR). CKD affects people all across the world; however, it is more common in underdeveloped nations. Meanwhile, slowing the course of CKD requires early identification. However, due to the high expense of detecting the condition and inadequate healthcare infrastructure, people in developing nations have not benefited from early-stage CKD screening [6]. While the prevalence of CKD is said to be 13.4% worldwide, Sub-Saharan Africa is said to have a 13.9% prevalence. According to a different study, the highest frequency of CKD in Africa was in West Africa at 16%. Numerous studies have shown that CKD is more common in poorer nations. Notably, 1 in 10 people have CKD in South Asia, including Pakistan, India, Bhutan, Bangladesh, and Nepal [5]. As a result, numerous researchers have suggested ML-based approaches for the early recognition of CKD. Particularly in underdeveloped nations, these ML techniques may offer efficient, practical, and affordable computer-aided CKD diagnosis tools to enable early identification and intervention. Researchers have suggested various ways to diagnose CKD successfully using the CKD dataset available at the Kaggle public library/repository [7].

Machine learning (ML) is becoming more important in healthcare diagnostics as it permits complex analysis, reducing human error and increasing prediction accuracy [8–11]. The most accurate methods for detecting illnesses such as liver disease, tumors, diabetes, and heart problems are now thought to be machine learning algorithms and classifiers [12, 13]. The major objective of this study is to suggest and put into practice a model for predicting chronic renal disease using machine learning. The goal is to treat CKD patients precisely and effectively while reducing treatment costs.

### 1.1 Contribution

In this study, we developed a system for predicting chronic kidney disease using machine learning techniques based on demographic data. The main contributions made by this research are summarized below:

- Exploratory data analysis is used to evaluate and enhance the quality of dataset.
- Perform data transformation, preprocessing, and appropriate hyperparameter tuning to increase the robustness of the created model.
- Develop a prediction model utilizing logistic regression (LR), Naïve Bayes (NB), support vector machine (SVM), and decision tree (DT).
- Validate the proposed framework using various performance evaluation metrics, and compare the results with current studies.

## 2 Review and Related Literature

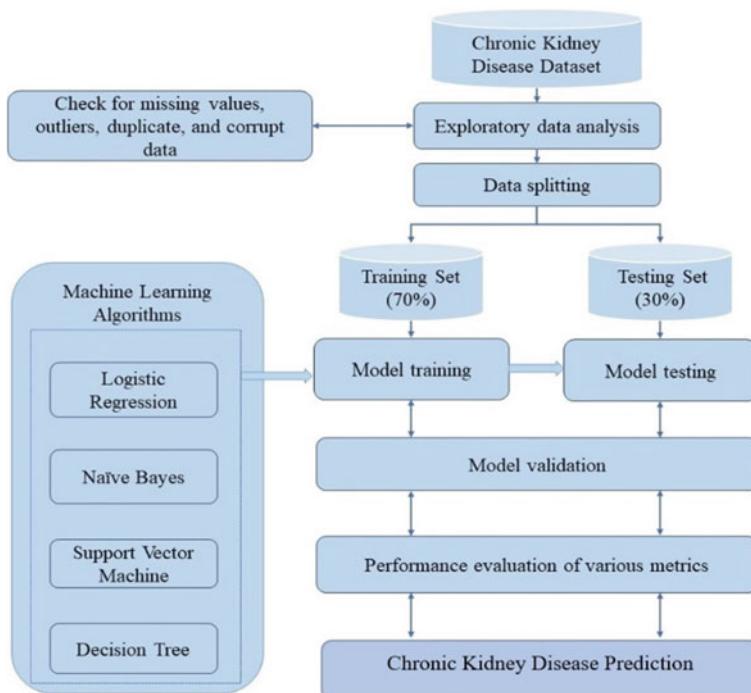
Numerous studies on the prediction of chronic renal disease have been carried out using machine learning. Below is a discussion of a few of them:

Bai et al. [14] used ML techniques for assessing the effectiveness in predicting the likelihood of end-stage kidney disease in patients with chronic kidney disease. In this investigation, logistic regression, Naïve Bayes, and random forest exhibited comparable predictability for the kidney failure risk equation. Islam et al. [15] developed a model to predicted risk factors for chronic kidney disease (CKD) and the onset of CKD. They employed numerous machine learning (ML) models, such as Naive Bayes, random forest, simple logistic regression, decision tree, and basic linear regression. The maximum accuracy was achieved by random forest, which performed better than other categorization methods (95, 94, and 93%, respectively). Debal et al. [16] used ML techniques including RF, SVM, and DT for model development. The results indicated RF performed well on recursive feature elimination method with k-fold cross-validation SVM and DT. Jhou et al. [17] aimed to create an effective hybrid important risk factor evaluation method for persons with MetS and stage III CKD using ML prediction models. They applied six ML models, but it was found that LR classifiers achieved the highest accuracy of 71.90% and RF achieved the lowest accuracy of 69.80%. Islam et al. [18] identified CKD at earlier using ML techniques. Previously specified input parameters are used to train and validate the models. Beginning with 25 factors in addition to the class attribute, this study eventually reduced the list to 30 of those parameters, which it found to be the most effective subset for identifying CKD. The XGBoost classifier demonstrated the best performance metrics, with accuracy, precision, recall, and F1-score values of 0.983, 0.98, and 0.98, respectively. Ye et al. [19] suggested that ML algorithms can be reliable resources for correctly predicting the likelihood of in-hospital death for CKD patients. In this paper, several machine/ensemble learning models are employed to

produce a better prediction framework for CKD. The gradient boosting decision tree machine (GBDT) model has attained a better predictive performance.

### 3 Proposed Methodology

The workflow of the proposed methodology to predict chronic kidney disease is shown in Fig. 1. Initially, the dataset on chronic kidney disease is imported, followed by the application of exploratory data analysis to enhance data quality assessment. Subsequently, the data is divided into training and testing sets, with 70% and 30% of the data allocated for training and validating/evaluating the models, respectively. Finally, the performance evaluation encompasses the calculation of various metrics for predicting obesity.



**Fig. 1** Methodology for prediction of chronic kidney disease

### ***3.1 Parameter Information***

The chronic kidney disease dataset used in this research was retrieved from the Kaggle public library/repository. The dataset is comprised of 400 patient records, where 250 are CKD and 150 are non-CKD patients with 24 independent attributes and a target variable. Table 1 provides the parameter information.

### ***3.2 Exploratory Data Analysis***

The optimal feature set has been identified based on their contribution towards the prediction of disease. The histogram of the selected parameters in the considered dataset is shown in Fig. 2, with the measurements of the parameters distributed along the  $x$ -axis and their computed values distributed along the  $y$ -axis. We utilized the Z-score and IQR score methods to screen for potential outliers in the dataset. The boxplot for considered parameters is depicted in Fig. 3. The correlation coefficient matrix (CCA) between the dependent and independent attributes is shown in Fig. 4.

## **4 Results and Discussion**

We used machine learning techniques like logistic regression, naïve bayes, support vector machine, and decision tree for disease prediction. We have explored 70% of data for training purpose and 30% of data to validate/test the models. Using several statistical and machine learning data, we validated the proposed model. In the next subsections, the related results are extensively presented and discussed.

### ***4.1 Confusion Matrix***

To compare the predicted results to the actual labels of a set of data, the effectiveness of a binary two-dimensional classification model is summarized using a confusion matrix. The confusion matrices for the LR, NB, SVM, and DT training and testing datasets are displayed in Figs. 5, 6, 7, and 8. According to the figures, DT and SVM performed better during the training phase and testing phase than NB and LR, but each of these groups had good outcomes overall. Based on the confusion matrices, we have assessed several measures, including accuracy, precision, recall, specificity, F1-score, and non-predicted values.

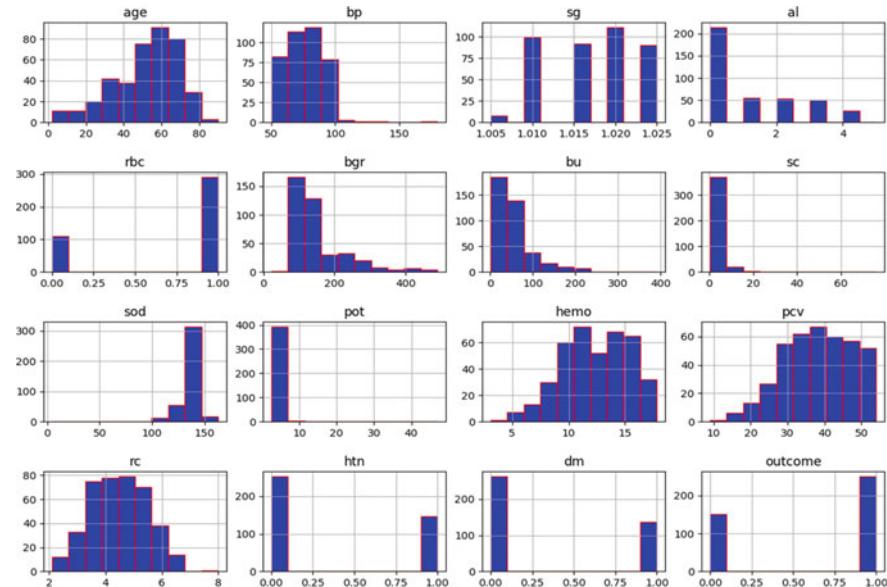
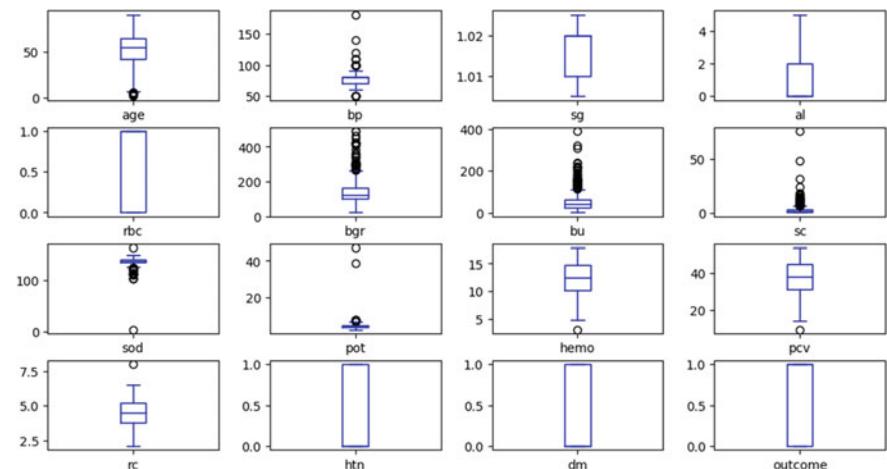
**Table 1** Parameter information of dataset

Attribute	Description	Measurement	Normal/range value
Age	Age of the person	Years	20–180 years
BP	Blood pressure of the person	Mm/Hg	80–120 mm/Hg
SG	Specific gravity of the person's body	(1.005, 1.010, 1.015, 1.020, 1.025)	—
Albumin	Albumin of the person	(0, 1, 2, 3, 4, 5)	3.4–5.4 g/dL
Sugar	The sugar level of the person	(0, 1, 2, 3, 4, 5)	90–130 mg/dL
Red blood cells	Red blood cells in the person's blood	(Normal, abnormal)	3.8–5.9 × 10 * 12/L
Pus cells	Pus cells in the person's body	(Normal, abnormal)	0–5 pus cells/HPF
Pus cells clumps	Pus cells clumps in the person's body	(Present, not present)	4–7 cells/HPF
Bacteria	Bacteria in human body	(Present, not present)	1–3% of the body mass
Blood glucose random	Blood glucose random in a person's body	Mgs/dl	140 mg/dL or below
Blood urea	Blood urea in the person	Mgs/dl	5–20 mg/dL
Serum creatinine	Serum creatinine in the person's body	Mgs/dl	0.7–1.3 mg/dL
Sodium	Sodium in the person's body	mEq/L	136–145 mEq/L
Potassium	Potassium in the person's body	mEq/L	3.5–5.2 mEq/L
Hemoglobin	Hemoglobin in the person's blood	Gms	12–18 g/dL
Packed cell volume	Packed cell volume in the person's body	—	2.5–97.5%
White blood cell count	WBCs in the person's blood	Cells/cumm	4.5–11.0 × 10*9/L
Red blood cell count	RBCs in the person's body	Millions/cumm	3.8–5.9 × 10*12/L
Hypertension	Hypertension level in a person's body	(Yes, no)	90–140 mm/Hg
Diabetes mellitus	Diabetes mellitus in the person	(Yes, no)	126 mg/dL or higher
Coronary artery disease	Coronary artery disease in the person	(Yes, no)	75–199 mg/dL
Appet	Appet in the person	(Good, poor)	—
Pedal edema	Pedal edema in the person	(Yes, no)	—

(continued)

**Table 1** (continued)

Attribute	Description	Measurement	Normal/range value
Anemia	Anemia in the person	(Yes, no)	11.6–16.6 g/dL
Class	Person's class	(CKD, non-CKD)	0–1

**Fig. 2** Histogram of selected parameter of the dataset**Fig. 3** Boxplot of selected parameter of the dataset

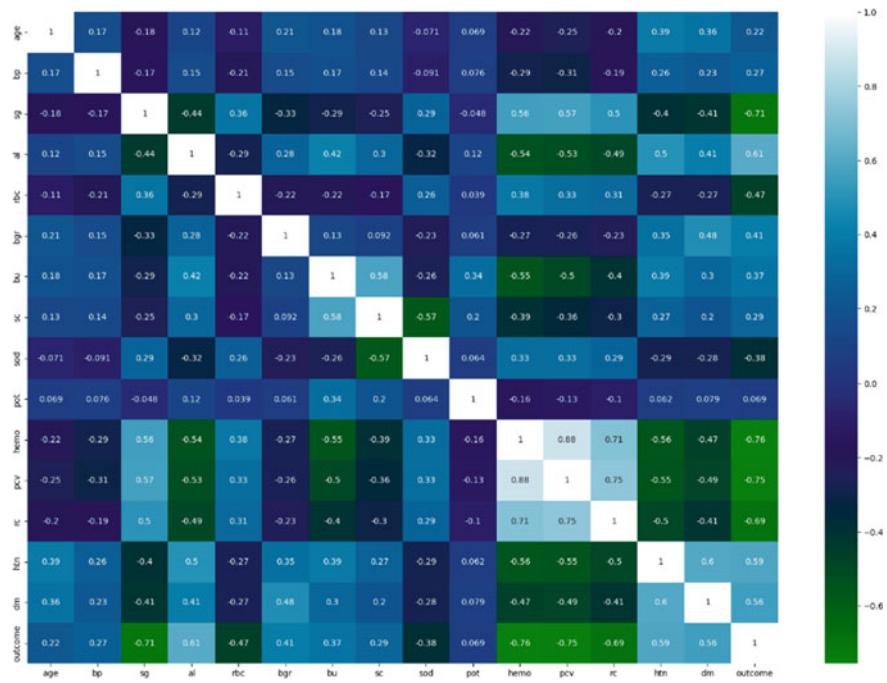


Fig. 4 Correlation coefficient matrix for selected parameter of the dataset

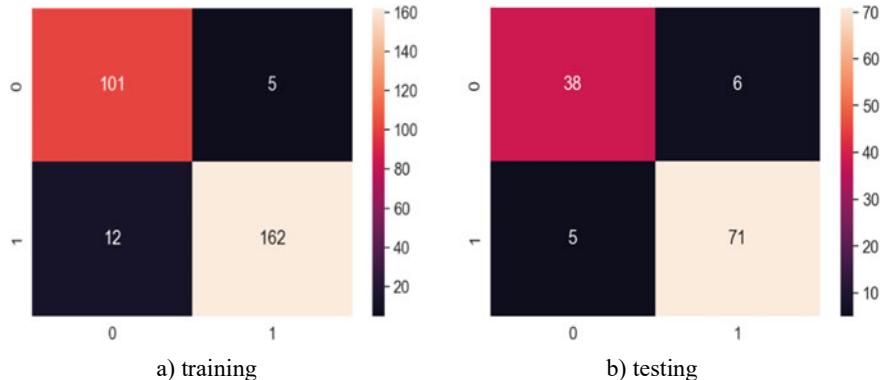
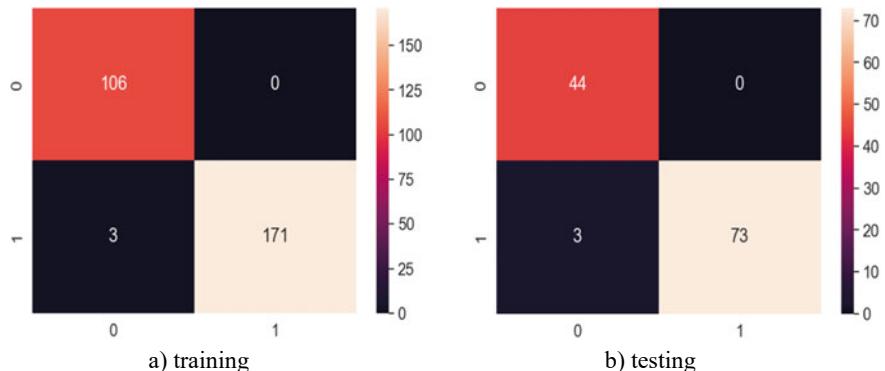


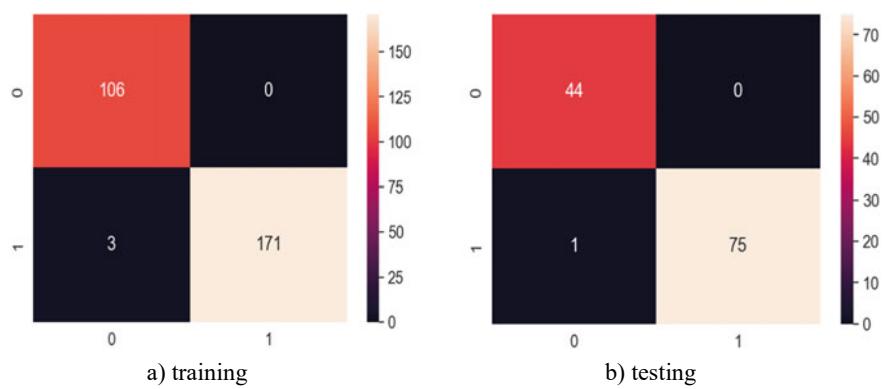
Fig. 5 Confusion matrix using LR, **a** training and **b** testing

#### 4.2 Accuracy of the Models

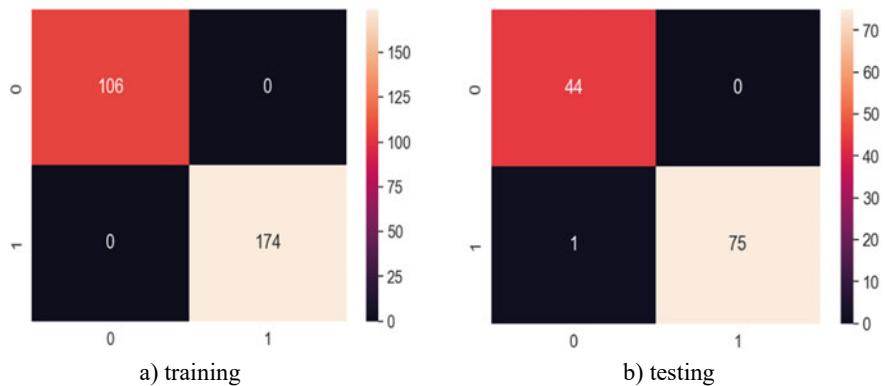
According to our findings, the accuracy of the decision tree model is the highest at 100% while training and 99.17% during testing, followed by support vector machine



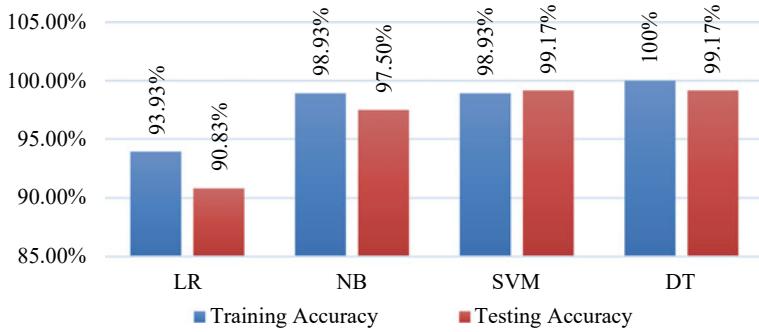
**Fig. 6** Confusion matrix using NB, **a** training and **b** testing



**Fig. 7** Confusion matrix using SVM, **a** training and **b** testing



**Fig. 8** Confusion matrix using DT, **a** training and **b** testing

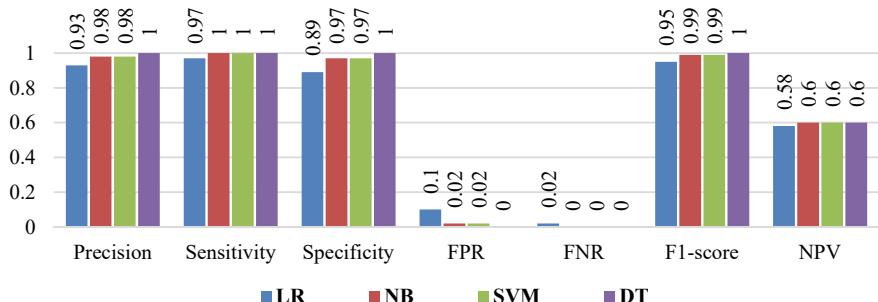


**Fig. 9** Training and testing accuracy of considered algorithms

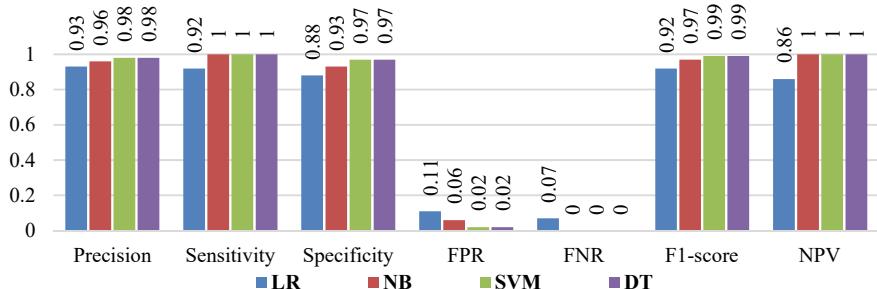
algorithm at 98.93% while training and 99.17% during testing, the naive bayes algorithm at 98.93% while training and 97.50% during testing, and logistic regression algorithm at 93.93% while training and 90.83% during testing, respectively, as shown in Fig. 9.

#### 4.3 Other Statistical Measurements

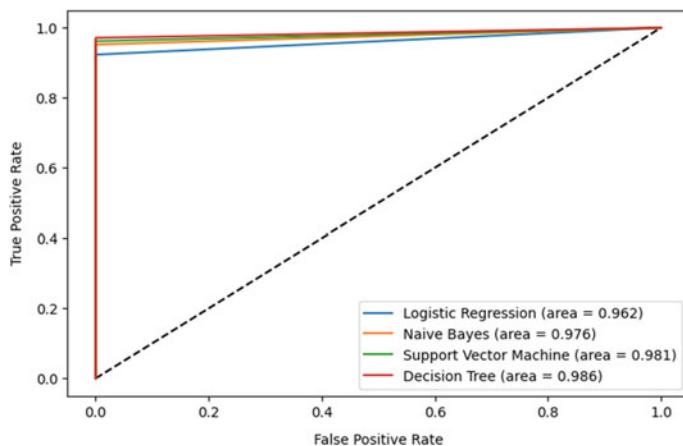
To develop and evaluate the performance of considered models, statistical/machine learning measurements like sensitivity, precision, specificity, false positive rate, false negative rate, F1-score, and negative predicted value (NPV) are calculated as shown in Figs. 10 and 11. It can be observed that DT and SVM display the best results for all metrics in both training and testing, but LR performs the worst in most situations.



**Fig. 10** Other performance metrics of algorithms for training set



**Fig. 11** Other performance metrics of algorithms for testing set



**Fig. 12** AUC-ROC curves for the considered models

#### 4.4 AUC-ROC Curve

The AUC-ROC curve shown in Fig. 12 was used to show the prediction ability of the considered machine learning models at different thresholds. It represents a false positive rate (FPR) versus true positive rate (TPR) along the  $x$ -axis and  $y$ -axis respectively.

### 5 Comparative Analysis

Our model yields better result when compared to the other research works, in which the authors explored data from various sources and conducted the research to predict chronic kidney disease. Table 2 shows the comparison based on techniques, dataset, and accuracy.

**Table 2** Comparative analysis with similar research work

Paper	Techniques used	Dataset used	Highest accuracy
[15]	Random forest, Naive Bayes, simple linear regression, simple logistic regression, decision stump, and linear regression model	A real-time CKD dataset with 1032 patient records and 15 attributes was gathered from a reputable medical college in Bangladesh	Random forest algorithm with 98.89% accuracy
[20]	Naive Bayes, multilayer perceptron (MLP), quadratic discriminant analysis (QDA), support vector machine, K-nearest neighbor, logistic regression, decision tree, and random forest	CKD dataset from UCI machine learning repository	Random forest with 99.75% accuracy
[16]	Extreme gradient boosting (XGBoost), decision tree, random forest, and support vector machine	The CKD dataset was collected between 2018 and 2019 from St. Paul's Hospital in Addis Ababa, Ethiopia	Support vector machine with 99.80% accuracy
[2]	Gradient boosting, Gaussian Naive Bayes, random forest, and decision tree	CKD dataset from UCI machine learning repository	Gradient boosting with 99% accuracy
[21]	Support vector machine, random forest, bagging (bootstrap aggregation), and K-nearest neighbor	CKD dataset from UCI machine learning repository	KNN with 99.50% accuracy
Our proposed work	Logistic regression, Naïve Bayes, support vector machine, and decision tree	CKD dataset from UCI machine learning repository	Decision tree with 100% accuracy while training and 99.17% during testing

## 6 Conclusion

Machine learning techniques are frequently used in the healthcare sector for better disease diagnosis and prediction. In this study, exploratory data analysis was used to assess the quality of the dataset under examination. To predict chronic kidney disease, we employed the four ML algorithms. The results were evaluated using various performance metrics. The experiment findings demonstrated that DT algorithm has achieved the best accuracy rate of 100% during training and 99.16% during testing, while LR algorithm had the lowest accuracy rate of 93.92% during training and 90.83% during testing. The decision tree also fared well in other parameters such as precision, recall, specificity, F1-score, and negative predicted value.

This work may be used with additional healthcare datasets that have comparable qualities or properties to widen the scope of this study. The proposed models could also be used by adding more sophisticated parameters like genetic data and biomarkers.

## References

1. Revathy S, Bharathi B, Jeyanthi P, Ramesh M (2019) Chronic kidney disease prediction using machine learning models. *Int J Eng Adv Technol* 9(1):6364–6367. <https://doi.org/10.35940/ijeat.A2213.109119>
2. Khalid H, Khan A, Zahid Khan M, Mehmood G, Shuaib Qureshi M (2023) Machine learning hybrid model for the prediction of chronic kidney disease. *Comput Intell Neurosci* 9266889. <https://doi.org/10.1155/2023/9266889>
3. Kalantar-Zadeh K, Jafar TH, Nitsch D, Neuen BL, Perkovic V (2021) Chronic kidney disease. *The Lancet* 398(10302):786–802. [https://doi.org/10.1016/S0140-6736\(21\)00519-5](https://doi.org/10.1016/S0140-6736(21)00519-5)
4. Qin J, Chen L, Liu Y, Liu C, Feng C, Chen B (2020) A machine learning methodology for diagnosing chronic kidney disease. *IEEE Access* 8:20991–21002. <https://doi.org/10.1109/ACCESS.2019.2963053>
5. Kovacs CP (2022) Epidemiology of chronic kidney disease: an update 2022. *Kidney Int Suppl* 12(1):7–11. <https://doi.org/10.1016/j.kisu.2021.11.003>
6. Ebaredoh-Mienye SA, Swart TG, Esenogho E, Mienye ID (2022) A machine learning method with filter-based feature selection for improved prediction of chronic kidney disease. *Bioengineering* 9(8). <https://doi.org/10.3390/bioengineering9080350>
7. Ifraz GM, Rashid MH, Tazin T, Bourouis S, Khan MM (2021) Comparative analysis for prediction of kidney disease using intelligent machine learning methods. *Comput Math Methods Med* 2021:6141470. <https://doi.org/10.1155/2021/6141470>
8. Malik MB, Ganie SM, Arif T (2022) Machine learning techniques in healthcare informatics: showcasing prediction of type 2 diabetes mellitus disease using lifestyle data. In: Roy S, Goyal LM, Balas VE, Agarwal B, Mittal M (eds) Predictive modeling in biomedical data mining and analysis. Academic Press, pp 295–311. <https://doi.org/10.1016/B978-0-323-99864-2.00001-9>
9. Babu S, Anil Kumar D, Siva Krishna K (2023) Intelligent multiple diseases prediction system using machine learning algorithm. In: Kumar R, Pattnaik PK, Tavares JMRS (eds) Next generation of internet of things. Springer Nature Singapore, Singapore, pp 641–652
10. Deepanshu, Singh K, Dhawan S (2022) Diagnosing multiple chronic diseases based on machine learning techniques: review, challenges and futuristic approach. In: 2022 international conference on communication, computing and internet of things, IC3IoT 2022—proceedings. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/IC3IOT53935.2022.9767930>
11. Ganie SM, Malik MB, Arif T (2021) Early prediction of diabetes mellitus using various artificial intelligence techniques: a technological review. *Int J Bus Intell Syst Eng* 1(4):325. <https://doi.org/10.1504/ijbise.2021.122759>
12. Arumugam K, Naved M, Shinde PP, Leiva-Chauca O, Huaman-Osorio A, Gonzales-Yanac T (2023) Multiple disease prediction using machine learning algorithms. *Mater Today Proc* 80:3682–3685. <https://doi.org/10.1016/j.matpr.2021.07.361>
13. Raheja V, Shah V, Shetty M, Patel P, Tiwari M (2022) Multi-disease prediction system using machine learning. In: 2022 international conference on futuristic technologies (INCOFT), pp 1–6. <https://doi.org/10.1109/INCOFT55651.2022.10094382>
14. Bai Q, Su C, Tang W, Li Y (2022) Machine learning to predict end stage kidney disease in chronic kidney disease. *Sci Rep* 12(1):8377. <https://doi.org/10.1038/s41598-022-12316-z>

15. Islam MA, Akter S, Hossen MS, Keya SA, Tisha SA, Hossain S (2020) Risk factor prediction of chronic kidney disease based on machine learning algorithms. In: 2020 3rd international conference on intelligent sustainable systems (ICISS), pp 952–957. <https://doi.org/10.1109/ICISS49785.2020.9315878>
16. Debal DA, Sitote TM (2022) Chronic kidney disease prediction using machine learning techniques. *J Big Data* 9(1):109. <https://doi.org/10.1186/s40537-022-00657-5>
17. Jhou MJ, Chen MS, Lee TS, Te Yang C, Chiu YL, Lu CJ (2022) A hybrid risk factor evaluation scheme for metabolic syndrome and stage 3 chronic kidney disease based on multiple machine learning techniques. *Healthcare (Switzerland)* 10(12). <https://doi.org/10.3390/healthcare10122496>
18. Islam MdA, Majumder MdZH, Hussein MdA (2023) Chronic kidney disease prediction based on machine learning algorithms. *J Pathol Inform* 14:100189. <https://doi.org/10.1016/j.jpi.2023.100189>
19. Ye Z et al (2023) The prediction of in-hospital mortality in chronic kidney disease patients with coronary artery disease using machine learning models. *Eur J Med Res* 28(1):33. <https://doi.org/10.1186/s40001-023-00995-x>
20. Nishat MM et al (2021) A comprehensive analysis on detecting chronic kidney disease by employing machine learning algorithms. *EAI Endorsed Trans Pervasive Health Technol* 7(29). <https://doi.org/10.4108/eai.13-8-2021.170671>
21. Ullah Z, Jamjoom M (2023) Early detection and diagnosis of chronic kidney disease based on selected predominant features. *J Healthc Eng* 2023:3553216. <https://doi.org/10.1155/2023/3553216>

# Designer Face Mask Detection Using Marker-Based Watershed Transform and YOLOv2 CNN Model



Arpita Vyas and Jankiballabh Sharma

**Abstract** Face mask detection using artificial intelligence (AI) has become more challenging due to high variability in modern designer masks such as single color, multicolor, textile, and printed. This paper presents a designer face mask detection using marker-controlled watershed transform and YOLOv2 CNN model. In the first stage, the face images are preprocessed and segmented using marker-controlled watershed transform by setting mask color as foreground and face color as background marker. The segmented image is applied to the YOLOv2 CNN model for the detection of the face mask. Marker-controlled watershed transform is employed for segmentation and highlights the multicolor mask texture, to improve the classification efficiency of the YOLOv2 CNN model. Simulation performed using different types of designer masks gives an accuracy of 86.66% and an F1-score of about 0.91, which verifies the efficiency of the proposed scheme. The technique deployed in this paper can be used to develop automated systems for face mask detection, classification, and alarm systems.

**Keywords** Face mask detection · YOLOv2 · Watershed segmentation

## 1 Introduction

Health experts recommend wearing face masks as a preventive and effective measure to prevent the spread of various airborne and communicable diseases. Numerous studies recommend wearing face masks even if one does not feel ill [1]. Various educational campaigns on face masks are conducted, and calls to “NO MASK, NO ENTRY” are made in public places such as shopping malls and movie theaters. The “Face Mask Detection System” is a lifesaver for us in this COVID-19 circumstance

---

A. Vyas (✉) · J. Sharma

Department of Electronics Engineering, Rajasthan Technical University, Kota, India  
e-mail: [arpita.20mtdc800@rtu.ac.in](mailto:arpita.20mtdc800@rtu.ac.in)

J. Sharma  
e-mail: [jbsharma@rtu.ac.in](mailto:jbsharma@rtu.ac.in)

because it is impossible to watch everyone in huge organizations and crowded settings to determine whether a person is wearing a mask or not [2]. The identification of the face is the first stage in determining whether a mask is present. This separates the entire procedure into two phases: face detection and mask detection. Face mask detection is greatly impacted by computer vision and image processing [3]. Face mask identification is a quite challenging task for the face detector models that are currently being offered. This is because different alterations, degrees of occlusion, and kinds of masks are evident on mask-wearing faces. Self-focusing, human-computer interaction, and image database management are all made easier with their help [1, 4]. The face mask detection model known as SSDMN2 was developed using deep neural network modules from OpenCV and TensorFlow, including a single-shot multibox detector object identification model [5].

The segmentation results produced by the watershed algorithm are reliable. Watershed segmentation based on marks and ore-marked region determination make up the two primary parts of the ore segmentation procedure. The watershed transformation's picture immersion method demands the shortest initial gradient position after marking. An efficient segmentation is the marker-based watershed method, which accurately divides the ore edge [6]. References [7–9] describe the watershed transform-based segmentation.

To overcome the issue of fast object detection, Yossip Redmon and colleagues suggested a novel method for object recognition that can produce real-time outcomes in 2016. You Only Look Once (YOLO) was the name given to it [10]. The object recognition challenge was viewed by the authors as a regression issue. Direct predictions of the bounding boxes and class probabilities were made using CNN. The confidence and category of the object are contained in the bounding box. The YOLO concept was applied to a single neural network. The object's coordinates were improved by adding a linear unit to the bounding box's location. The classification's accuracy improved as a result. They improved several parts of the network structure while also borrowing concepts from various optimization techniques, creating YOLOv2. YOLOv2 is the state of the art for standard recognition tasks. It can be run at different sizes and provides a good balance between speed and accuracy [11–14].

In the detection and identification of objects in digital photographs, such as pedestrian detection and crowd counting, convolutional neural network (CNN) is a deep learning image classification technology that is extensively utilized. Researchers frequently employ the CNN method, which is regarded as the finest algorithm for object identification and object recognition. CNN does have one disadvantage, though, which is the requirement for capable hardware performance throughout the training phase, just like other deep learning models. Deep learning has demonstrated that it is effective based on recent performance. The advancement of computing, which is continually moving toward better, larger datasets, and deeper network architecture, has an impact on this [4, 15–17].

The rest of the paper is arranged as preliminaries in Sect. 2, and the proposed work is described in Sect. 3. Results are discussed in Sect. 4, and Sect. 5 concludes the presented work.

## 2 Preliminaries

### 2.1 Marker-Controlled Watershed Segmentation

Typically, the watershed method results in over segmentation. Use of the marker concept can help to solve this issue. The two types of markers are internal markers and external markers. External markers are important for identifying background, whereas internal markers are useful for identifying the things of interest. Markers are used to apply the gradient of the image and are viewed as beginning points. In order to produce better segmentation results, the regions containing the markers are considered as discrete watersheds in the gradient image and have the watershed algorithm applied to them. Image segmentation is the process of dividing up the regions of interest in an image into groups that adhere to a predetermined threshold or standard in order to improve the outcomes of visual analysis or other post-processing. Edge-based and region-based image segmentation techniques are two of the traditional methods utilized in watershed-based segmentation [6, 18].

### 2.2 YOLOv2

YOLOv2 is the starting point for standard recognition tasks that can be run in different sizes and offer a good balance between speed and accuracy. A real-time object recognition system is called You Only Look Once (YOLO). DarkNet-19 is the basic building block of YOLOv2, its architecture contains nineteen convolution layers and five max-pooling layers, and serves as the fundamental model for YOLOv2. Although much faster, it is less precise. As a result, the network is made deeper by including more convolutional layers. There are 27 layers altogether. About  $13 \times 13$  grids are used to divide up the images that are delivered to the YOLOv2 network. The ultimate output of the network is thirteen by thirteen by n; here, n is the quantity of filters in the last layer. These filters outline the number of classes that can be anticipated. As the network progresses from its 18th layer to its final output layer, convolutional layers and max-pooling layers convert the image into a  $13 \times 13$  format [15].

An  $S \times S$  grid is produced by YOLO using the input image. When the center of an object passes across a grid cell, that cell takes ownership of that object and calculates the B-bounding boxes and confidence intervals for those boxes. The model's level of certainty that an item is present in the box as well as its predictive accuracy is included in the confidence rating. The complex object recognition problem is transformed into a regression model via the YOLO design, which allows for real-time speed object identification and end-to-end training. This not only reduces the complexity of model training but also considerably improves the execution efficiency of the entire recognition model, enabling the CNN-based object recognition model to achieve a breakthrough in runtime [4].

### 3 Proposed Work

The suggested face mask identification method is discussed in this section and is based on CNN, YOLOv2, and image segmentation using marker-based watershed segmentation. Figure 1 displays the proposed method's elaborate block diagram.

The summarized steps of the proposed method are given below.

#### A. Dataset Collection

The masked face dataset from Kaggle was the source of the data for the present research. The images assembled are images of a bunch of people's faces. The faces that use masks (wm) and the faces that do not use masks (wom) are categorized into two groups in the images. There were 120 total photographs used in this study, including 30 images in the wom class and 90 in the wm class.

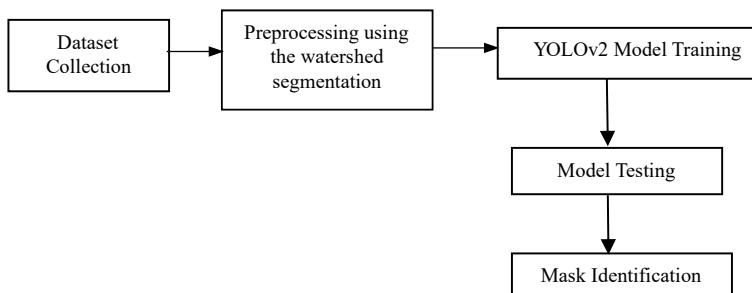
#### B. Preprocessing with Watershed Segmentation

Now watershed segmentation is applied to the input images. Segmentation with the watershed transformation works better if you can identify or "mark" objects in the foreground and background. Watershed segmentation helps to easily identify different mask types including colored and printed masks. Steps in watershed segmentation:

- Firstly, take any color image as the input and then converting it to grayscale.
- Secondly, using gradient size as the segmentation function.
- Then marking the items in the foreground.
- Calculating the background markers.
- Calculating the watershed transformation of the segmentation function.
- Visualize the result.

#### C. Model Training with YOLOv2

A real-time object identification system called You Only Look Once (YOLO) was developed. Convolutional neural network regions (Faster R-CNNs), another two-stage deep learning object detector, are slower than YOLOv2.



**Fig. 1** Block diagram of proposed scheme

The YOLOv2 model works on three main steps:

- Firstly, the image is separated into several grids. There are many equal-sized grids present with each cell grid of  $S \times S$  size. Every grid cell will be able to detect items that enter it. For instance, a grid cell will be in charge of detecting an object if its center appears within that cell.
- A bounding box is another term for an outline that highlights an object in an image. Each bounding box in the picture possesses the qualities listed below: sizes in both  $b_w$  and  $b_h$ . Class, which encompasses objects like person, car, traffic light, etc., is represented by the letter C. Box center ( $b_x, b_y$ ) bounding.
- Thirdly, intersection over union causes the projected bounding boxes to be equal to the actual boxes of the objects. This phenomenon results in the elimination of bounding boxes whose height and width do not correspond to the object dimensions. Special bounding boxes designed specifically to fit the items will comprise the final detection.

To generate network predictions from an input image, the YOLOv2 model employs a deep learning CNN. The predictions are decoded by the object detector, which also produces bounding boxes. The YOLOv2 process is broken down into the following steps:

### **Step 1: Design of the YOLOv2 network layers**

Layer by layer, we created a unique YOLOv2 model for this model. The input layer is where the model is created. Conv, Batch Norm, and ReLU layers are then found within the detection subnet, followed by the transform and output layers, represented by the YOLOv2 transform layer and YOLOv2 output layer objects. The CNN output is converted by the YOLOv2 transform layer into the format needed for object detection. The anchor box parameters and the loss function used to train the detector are both implemented in the YOLOv2 output layer.

The **image input layer** function is used to define the image input layer. Set the filter size of the convolution layer to [3]. This size is common in CNN architectures.

The size (height and width) of the local regions that the neurons in the input connect to is defined by the filter size.

filter size = [3 3]; input layer = image input layer ([225 225 3]);

We employed a repeated series of the middle layers: Convolution2dLayer, Batch Normalization Layer, ReLU Layer, and Max-Pooling Layer, in accordance with the basic methodology described in the YOLO9000 paper.

### **Step 2: Create a layer graph for the YOLOv2 network**

To process the layers, combine the first and middle layers and turn them into a layer graph object. Using the input data, calculate the multiple classes.

### **Step 3: Define anchor boxes**

A group of bounding boxes with predetermined height and breadth are called “anchor boxes”. These boxes are frequently selected in accordance with the object sizes in

your training datasets and are meant to represent the scale and aspect ratio of particular object classes you want to recognize. Using anchor boxes, a network may identify numerous objects, objects of different sizes, and objects that are overlapping. The size and scale of the objects in the training data are used to choose the anchor boxes.

#### **Step 4: Assembling the yolov2 network**

The YOLOv2Layers function builds a YOLOv2 network that simulates the YOLOv2 object detector's network architecture. The YOLOv2 object detection subnetwork receives its input from the features collected from this layer.

#### **Step 5: Train the network**

“TrainingImageSize”—An M-by-2 matrix specifying one or more training image sizes. At each epoch, the training images are randomly changed to one of the specified sizes. Specify more than one size to facilitate detection of objects in images of different sizes. On the basis of the ground truth data and training images used with the trainYOLOv2ObjectDetector function, the YOLOv2 object detector recognizes certain objects in images. Pass the YOLOv2 object detector to the detect object function to find objects in an image.

```
detector = trainYOLOv2ObjectDetector(trainingData, graph, options);
```

A GPU coder can be used to create code for the YOLOv2ObjectDetector after the detector has been trained and assessed.

#### **D. Testing the model**

The trained architectures were assessed for calculation accuracy, precision, recall, F1-score, and specificity during the model testing phase.

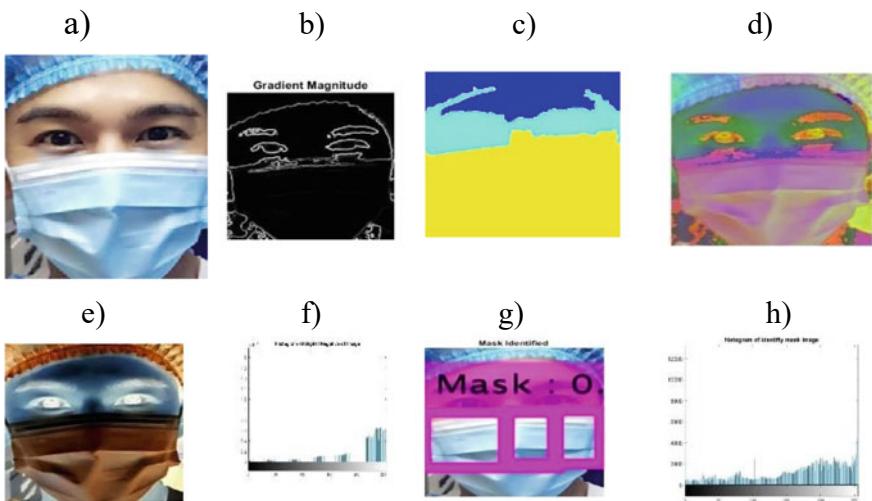
## **4 Results**

The presented technique's simulation results are demonstrated in this section using the MATLAB R2023a program. To display the visual findings, we chose six images at random from the dataset. We used a code, *wm*, for photographs with a mask and *wom*, for images without a mask, to represent the images in the collection simply.

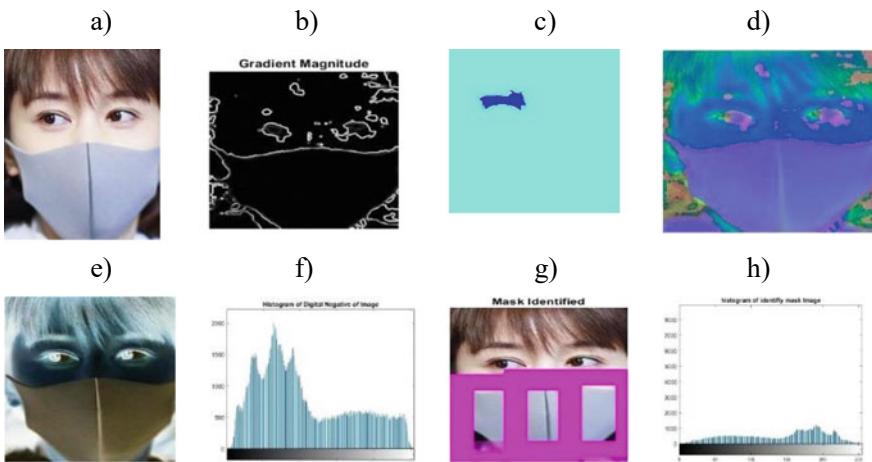
The selected images are combined images of different types of masks such as surgical masks, colored and multicolored textiles, and printed masks, i.e., *wm26*, *wm46*, *wm78*, *wm81*, *wm83*, and *wm87*.

The results of six images are shown in Figs. 2, 3, 4, 5, 6, 7, and 8 in which Figs. 2a, 3a, 4a, 5a, 6a, 7a, and 8a show the original image from the dataset. As the parts of watershed segmentation, gradient magnitude is shown in Figs. 2b, 3b, 4b, 5b, 6b, 7b, and 8b. The colored watershed label image is shown in Figs. 2c, 3c, 4c, 5c, 6c, 7c, and 8c. Digital negative image is shown in Figs. 2e, 3e, 4e, 5e, 6e, 7e, and 8e. Now histogram of the digital negative image is shown in Figs. 2f, 3f, 4f, 5f, 6f, 7f, and 8f.

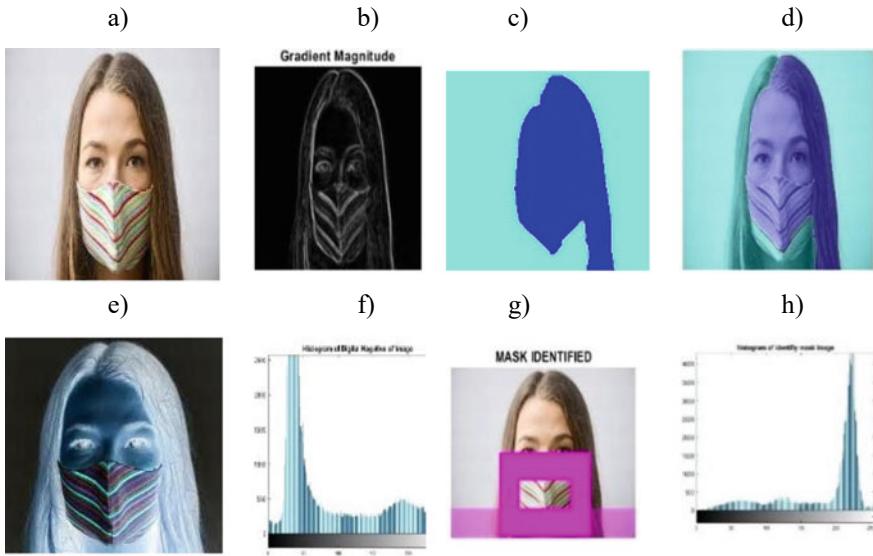
Mask-identified image is shown in Figs. 2g, 3g, 4g, 5g, 6g, and 7g, and histogram of mask-identified image is shown in Figs. 2h, 3h, 4h, 5h, 6h, and 7h.



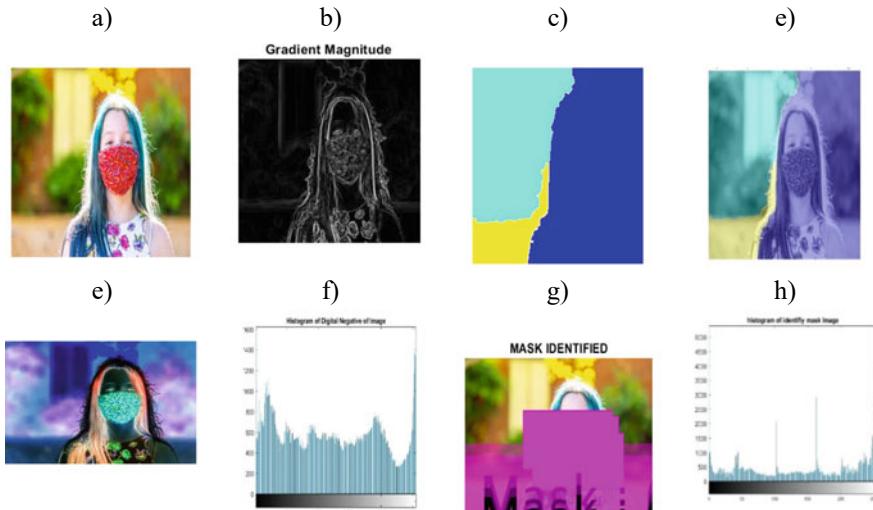
**Fig. 2** Image wm46 from dataset, i.e., **wm stands for with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



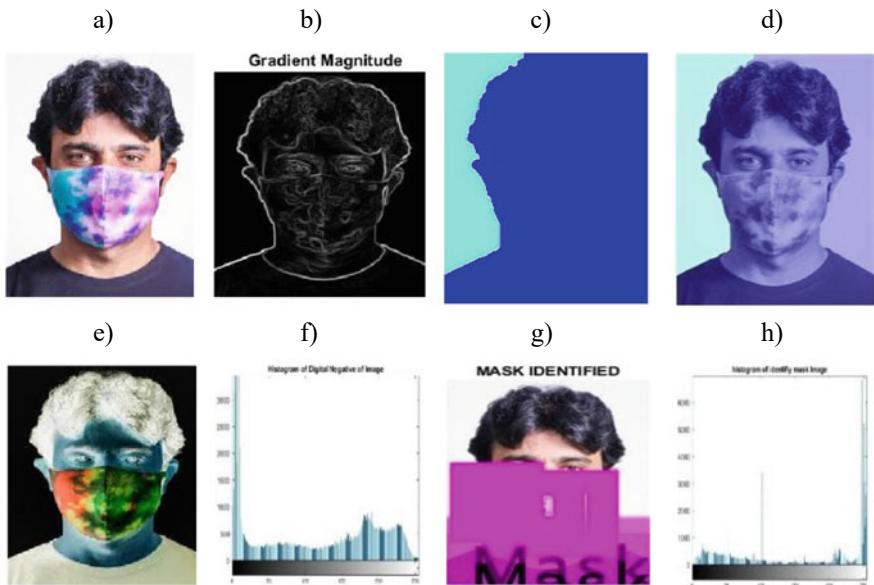
**Fig. 3** Image wm26 from dataset, i.e., **wm stands for with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



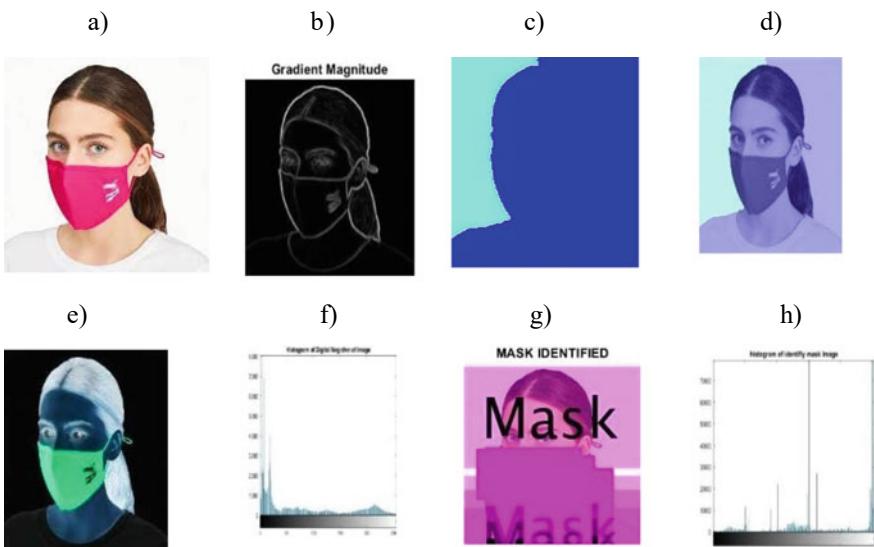
**Fig. 4** Image *wm78* from dataset, i.e., **wm stands for with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



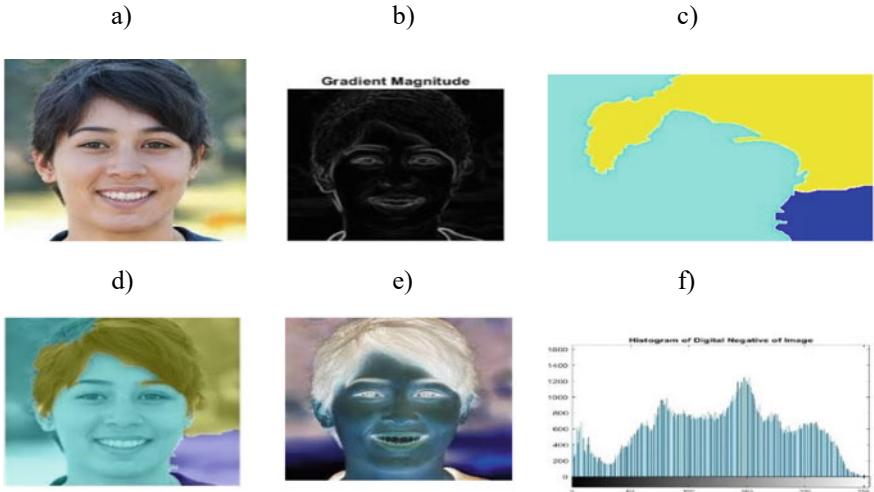
**Fig. 5** Image *wm83* from dataset, i.e., **wm stands for with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



**Fig. 6** Image *wm81* from dataset, i.e., **wm** stands for **with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



**Fig. 7** Image *wm87* from dataset, i.e., **wm** stands for **with mask**. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image, **g** mask-identified image, **h** histogram of identified image



**Fig. 8** Image wom8 from dataset, i.e., **wom** stands for without mask. **a** Original image, **b** gradient magnitude image, **c** colored watershed image, **d** superimposed image, **e** digital negative image, **f** histogram of digital negative image

### A. Single Color Masks

Mask-identified image and histogram of mask-identified image of Fig. 8 will not be shown as the mask is not present.

Now, these research results are combined to form confusion matrix that is displayed in Table 1. Table 1 provides the values of TP, FP, TN, and FN, which stand for true positive, false positive, true negative, and false negative, respectively. These confusion matrix values were used to generate the accuracy, precision, recall, F1-score, and specificity results that are displayed in Table 2.

### B. Images with Multicolor and Printed Masks

### C. Images with Textile Masks

### D. Images Without Masks

## 5 Confusion Matrix

Table 1 shows the values of TN, FN, FP, and TP in the form of a matrix known as the confusion matrix where  $N = 120$  is the total number of images in the dataset are 120. Calculations made using the confusion matrix are presented in Table 2 which shows accuracy, precision, recall, F1-score, and specificity.

**Table 1** Confusion matrix

Total images ( $N$ ) = 120	Predicted no	Predicted yes
Actual no	TN = 17	FP = 13
Actual yes	FN = 3	TP = 87

**Table 2** Calculations using confusion matrix

Accuracy	86.66%
Precision	87%
Recall	96.66%
F1-score	0.9127
Specificity	56.66%

## 6 Conclusions

In this paper, a method for detecting multicolored designer face masks using a marker-driven watershed transform and the YOLOv2 CNN model was presented. The simulations performed for various designer face masks such as multicolored, textile, and printed demonstrate how the suggested technique performs better than the standard ways and provide evidence to support the effectiveness of the approach. The proposed technique can be utilized to create automated systems for identifying and categorizing face masks in public areas. Our accuracy with the technique used in this research is 86.66%, and our F1-score is roughly 0.91.

## References

1. Vibhuti, Jindal N, Singh H, Rana PS (2022) Face mask detection in COVID-19: a strategic review. *Multimed Tools Appl* 81(28):40013–40042
2. Sharma A, Gautam R, Singh J (2023) Deep learning for face mask detection: a survey. *Multimed Tools Appl* 1–41
3. Goyal H, Sidana K, Singh C, Jain A, Jindal S (2022) A real-time face mask detection system using the convolutional neural network. *Multimed Tools Appl* 81(11):14999–15015
4. Ramadhan MV, Muchtar K, Nurdin Y, Oktiana M, Fitria M, Maulina N, Elwirehardja GN, Pardamean B (2023) Comparative analysis of deep learning models for detecting facemasks. *Procedia Comput Sci* 216:48–56
5. Nagrath P, Jain R, Madan A, Arora R, Kataria P, Hemanth J (2021) SSDMNV2: a real-time DNN-based face mask detection system using a singleshot multi-box detector and MobileNetV2. *Sustain Cities Soc* 66:102692
6. Manda MP, Park C, Oh B, Kim HS (2019) Marker-based watershed algorithm for segmentation of the infrared images. In: 2019 International SoC Design Conference (ISOCC), pp 227–228. IEEE
7. Peng C, Liu Y, Gui W, Tang Z, Chen Q (2021) Bubble image segmentation is based on a novel watershed algorithm with an optimized mark and edge constraint. *IEEE Trans Instrum Meas* 71:1–10

8. Zhang W, Jiang D (2011) The marker-based watershed segmentation algorithm of ore image. In: 2011 IEEE 3rd international conference on communication software and networks, pp 472–474. IEEE
9. Shylaja SS, Murthy KB, Natarajan S, Prasad A, Modi A, Harlalka S (2012) Feature extraction using marker-based watershed segmentation on the human face. In: 2012 international conference on computer communication and informatics, pp 1–5. IEEE
10. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779–788
11. Redmon J, Farhadi A (2017) YOLO9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7263–7271
12. Kwaghe OP, Gital AYU, Madaki AG, Abdulrahman ML, Yakubu IZ, Shima IS (2022) A deep learning approach for detecting face mask using an improved Yolo-V2 with SqueezeNet. In: 2022 IEEE 6th conference on information and communication technology (CICT), pp 1–5. IEEE
13. Seong S, Song J, Yoon D, Kim J, Choi J (2019) Determination of vehicle trajectory through optimization of vehicle bounding boxes using a convolutional neural network. Sensors 19(19):4263
14. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. PMLR, pp 448–456
15. Shamrat FJM, Chakraborty S, Billah MM, Al Jubair M, Islam MS, Ranjan R (2021) Face mask detection using convolutional neural network (CNN) to reduce the spread of COVID-19. In: 2021 5th international conference on trends in electronics and informatics (ICOEI), pp 1231–1237. IEEE
16. Rajeshkumar G, Braveen M, Venkatesh R, Shermila PJ, Prabu BG, Veerasamy B, Bharathi B, Jeyam A (2023) Smart office automation via faster R-CNN-based face recognition and the internet of things. Meas Sens 27:100719
17. Rekha V, Manoharan JS, Hemalatha R, Saravanan D (2022) Deep learning models for multiple face mask detection under a complex big data environment. Procedia Comput Sci 215:706–712
18. Fan H (2020) Application of improved watershed image segmentation algorithm in post-processing of capacitive tomographic images. In: 2020 IEEE international conference on artificial intelligence and information systems (ICAIIS), pp 485–489. IEEE

# Drought Prediction Using Machine Learning Forecasting Model in the Context of Bangladesh During 1981–2018



Alomgir Hossain, Momotaz Begum, and Nasim Akhtar

**Abstract** Drought is a short engagement in water or moisture availability substantially under the regular or expected. The amount for particular length drought occasions is recognized month-to-month. In this study, rainfall records from Bangladesh Meteorological Department Bangladesh with proved standard precipitation index (SPI) between 1981 and 2017 are used. Historical document of drought is received from Bangladesh Bureau of Statistics. The International Disaster Database may be used to validate the SPI result, and SPI is calculated at the District of Dinajpur in Bangladesh. The SPI can observe drought phenomena on a nearby scale. For this study, we have calculated SPI scale value for the dataset of rainfall like SPI month-1; month-3, month-6, month-9, and month-12, and we have used the short-term scale value for detecting the drought scale. We have calculated in this one sub-area throughout the country to detect the drought using shiny packages which is server based. In our study, we have observed that moderate drought frequency is excessive anywhere in the country. Based on the forecasting method FB Prophet which is also known as time series algorithm for forecasting, we have predicted the future year SPI value for the month which has given most moderate drought according to our study. Our main purpose is to be able to detect the drought of upcoming year so that we can be able to take necessary steps to prevent it. In our study, the forecasting methods are most accurate to predict the drought forecast for future.

**Keywords** Engagement · SPI · Dinajpur · Short-term scale · Sub-area · Time series forecast · FB Prophet

---

A. Hossain (✉) · M. Begum · N. Akhtar

DUET-Dhaka, University of Engineering & Technology, Gazipur, Bangladesh

e-mail: [alomgir.hossain@iubat.edu](mailto:alomgir.hossain@iubat.edu)

M. Begum

e-mail: [drmomotaz@duet.ac.bd](mailto:drmomotaz@duet.ac.bd)

N. Akhtar

e-mail: [drnasim@duet.ac.bd](mailto:drnasim@duet.ac.bd)

A. Hossain

IUBAT-International University of Business Agriculture and Technology, Dhaka, Bangladesh

## 1 Introduction

Drought is typically described as a deficiency of precipitation over a prolonged length of time (generally as Eason or greater), ensuing in a water shortage. Weather drought is an herbal phenomenon that repeats itself over a quick length of time [1]. Disasters due to loss of rainfall can bring about sizeable financial loss. However, the terrible consequences of meteorological droughts may be monitored and mitigated by achieving drought forecasts depending on how they are described and identified [2]. Bangladesh is one of the toughest hit nations through numerous climate screw pre-monsoon and post-monsoon droughts, tropical cyclones, and summer time season monsoon floods. Due to worldwide warming, South Asia has maximum of the weather a version for predicting a lower in precipitation with inside the dry season. It will increase seasonally and all through the monsoon season. This can reason a mixture of greater excessive droughts and floods in this area. According to the National Drought Report, Bangladesh already has a 2006 mitigation center. It suggests that the frequency of droughts has elevated in the latest years.

On the other hand, the meteorological drought is increasing daily, generating a big problem for our country's agriculture, water, and industry. Production relies upon on precipitation for essential detection of the consequences of massive precipitation variability. Research is required to identify the different drought levels for mitigating severe drought [3]. Drought turns into very essential for coverage making. Also today, meals safety is an essential worldwide issue [4]. Since drought is carefully associated with meals, mitigate the consequences of drought in Bangladesh.

In Bangladesh, numerous researches have tested the consequences of drought in agriculture, meals production, financial system, and society [5, 6]. Currently, thus far there may be no widespread drought index approach utilized in Bangladesh for drought diagnosis. Target current studies are to measure the drought according to their scale as year and predicted the future value, so that we can know about the drought from the beginning of the year. Drought styles are the usage of the same old drought index approach. Drought uses the SPI scale to detect whether they can be detected and use the forecasting method to forecast the future value for the scale.

## 2 Related Works

Based on the type of water insufficiency, such as precipitation, runoff, soil moisture, and water availability, respectively, droughts can be classified as agricultural, meteorological, hydrological, or socioeconomic. The most significant of these categories is meteorological drought which is seldom dependent on precipitation and gives rise to other drought categories when it persists for an extended period of time. To model meteorological, hydrological, and agricultural drought, a number of drought indices, such as the standardized precipitation index (SPI), Palmer drought severity index, standardized precipitation evapotranspiration index (SPEI), and standardized runoff

index have been developed in recent years. Because they are simple to use, adaptable, and can estimate drought over a wide range of time periods with minimum data requirements, standardized drought indices have been widely used in drought modeling [1].

Precipitation occurs at the specified automatic conversion rate when the cloud water threshold is exceeded. This threshold specification is based on empirical observations of the amount of cloud liquid water in clouds. This scheme contains simple formulas for raindrop accumulation and evaporation. In many modeling applications, it is crucial to precisely depict cloud operations. However, clouds are frequently underrepresented in both regional and global climate models (RCMs and GCMs, respectively), in part because some of the main cloud processes take place at scales that are too large or too small for existing models to resolve [2].

Predicting drought conditions for weeks to months in advance are less common, but would provide a more effective early warning system for improved drought response, mitigation, and adaptation planning. Many drought indices are implemented to monitor the current extent and status of drought, so stakeholders such as farmers and local governments can appropriately respond. Methods to forecast drought conditions weeks to months in advance are less common but would provide a more effective early warning system to enhance drought response, mitigation, and adaptation planning [7].

In this paper, they proposed a gadget called as Diagnosis of Drought in Bangladesh. In this gadget drought occasion is recognized from the Bangladesh Meteorological Department's month-to-month precipitation records for Bangladesh the usage of the standardized precipitation index (SPI) from 1961 to 1990 [3].

Little attention has been paid to drought mitigation and preparedness despite drought being a recurring phenomenon in Bangladesh. This article presents a method for spatially assessing drought risk in Bangladesh. A conceptual framework was used in this study to highlight the combined role of hazards and vulnerabilities in defining risk [4].

Droughts have delivered historical civilizations to their knees with famines, meals scarcity, and the lack of lives and property. Evidence shows that drought-affected region is growing globally. When it comes to spreading precipitation geographically, topography has a major role, but only in sub-regions with complicated topographical features (like the Alps). Resolution affects a number of other factors, including cloudiness, surface energy flows, and precipitation intensity distributions [5].

The natural variability that makes up drought includes a number of difficult-to-predict hydrologic processes, including precipitation, evaporation, soil moisture, and groundwater level. Effective strategies for predicting essential aspects of drought episodes are crucial for reducing the impact of drought. As a result, it is feasible to analyze and predict the drought indicators. Palmer drought severity index (PDSI), crop moisture index (CMI), and standard precipitation index (SPI) are three often used metrics as stand-alone drought indices among many other drought monitoring techniques [6].

Environmental catastrophes like droughts have a significant impact on agriculture, wildfires, and water supply. Different viewpoints have different markers for

describing drought and its severity, such as meteorological, socioeconomic, and ground water conditions. Drought monitor is an analysis of the drought in the USA that takes into account domain knowledge, climate indices, and model outputs. It is linked to the Palmer index, the standard precipitation index (SPI), and soil moisture, among other drought measures [8].

Millions of people in South Asia face difficulties as a result of drought, a climatic disaster that affects many different industries. In order to disclose the complexity of drought and the factors that contribute to it, accurate and thorough drought information combined with an effective monitoring system is crucial. Using the soil moisture deficit index (SMDI) as a response variable over three crop phenology stages, this study assessed agricultural drought and employed two machine learning models, distributed random forest (DRF) and gradient boosting machine (GBM), to better compare the performance of deep learning models [9].

In this research paper, for the first time they investigated the potential drought prediction model for Pakistan by machine learning approach. It was found that in the Rabi season, SPEI is positively correlated with relative humidity for Mediterranean Sea [10].

This article shows a multi-hazard susceptibility mapping framework using deep learning algorithm (CNN), and they found area of hazard prone. These are 16.14, 4.94, and 30.66% of susceptibility to flash floods, debris flows, and landslides, respectively [11].

Researcher mentioned the future research direction based on drought forecasting using web of science extracted data [12]. In this paper, they proposed using ML and SPEI to analysis of drought in the area of Tibetan Plateau, China, for the period of 1980–2019. After this analysis, the developed models produced satisfactory results [13].

This article represents a comprehensive analysis of different drought prediction model and ML, and DL is the best way to predict drought than traditional model [14]. This study investigated the use of ML methods for predicting the indicator over Sweden. They used three data arrangement methods: (multi-features, temporal, and spatial) [15].

### 3 Dataset and Proposed Method

In our study, we have used a local dataset as part of the data collection phase of the suggested research design from Bangladesh Bureau of Statistics. According to our proposed system, first of all, we have to check data type, and then based on the data type, we have to select our algorithm.

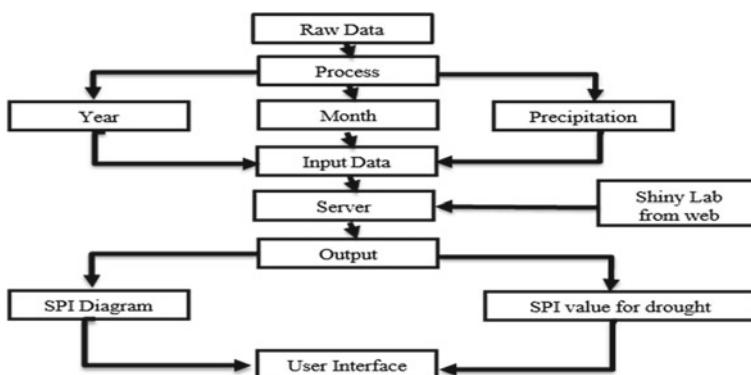
### 3.1 Proposed Model

In forecasting drought, we took the raw data for processing it and divided it into three parts. From Fig. 1, these are year, month, and precipitation. We have to take the precipitation value according to month-wise by adding whole month data which indicates days. After that, we make a file for the process data and give as input to our system, which match the data with the server library, process it, and give the output. The output value divided into two parts one is SPI diagram, and another one is SPI value; based on that, we can detect drought according to term-wise. From the training set, it will proceed to algorithm. After that, we will get some predicted value.

After calculating the SPI value for different months, the result is indicated for month-1 (SPI-1), month-3 (SPI-3), and month-6 (SPI-6). From the SPI scale, which one gives the most detected drought, we will take data as input in the times series algorithm. After that we will make sure that our data is fit to the algorithm, and then we will go for data processing. By processing the data, it will be divided into two parts: one is training set, and another one is testing set. From the training set, it will proceed to algorithm part where we will use FP Prophet algorithm. After that, we will get some predicted value, and then we will evaluate it with the test data.

It will give predicted value. After that we will evaluate it with the matrices value for better accuracy.

Enforce times series algorithm based on which short-term value gives more amount of drought. In our study, we basically work with Prophet of time series algorithm which has more efficiency than the other algorithm of times series. It is primarily based totally on an additive version where nonlinear tendencies are matched with yearly, weekly, and day by day seasonality, plus vacation effects. It is designed to handle missing data, large outliers, and multiple change points. Based on that, we fit the dataset in the algorithm part, and then we have calculated next 365 days forecast based on our dataset. Prophet also allows for prior information to be incorporated into



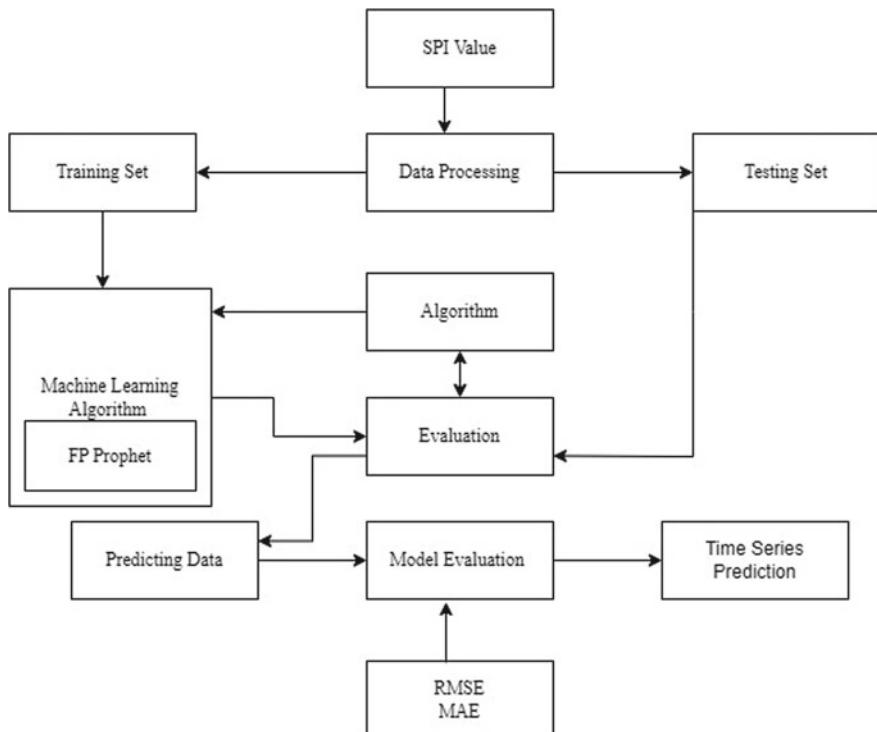
**Fig. 1** Process model of detecting SPI value

the model. This can be used to constrain the model's forecasts to be consistent with known information about the future or to incorporate domain-specific knowledge about the time series.

### 3.2 Data Processing

We have used six variables from our dataset which is split into two pairs: a training group with five hundred rows and a testing group with two hundred fifty rows. Figure 2 includes a visual representation of its dataset's attributes as well as its appropriate explanations.

Table 1 represents our dataset. We have calculated the rainfall amount according to month after addition of day's data. Then we have divided the data in year, month, and precipitation. After that, we have calculated the value of SPI of different months based on the scale where we have used five variables that have five hundred data. From that, we have measured the detection of drought, which drought scale gives



**Fig. 2** Process model of SPI value calculated for future

**Table 1** Visual representation of its datasets of SPI

Parameter	Type	Explanation
Year	Numerical	Time
Tmax	Numerical	Temperature max
Tmin	Numerical	Temperature min
Rainfall (mm)	Numerical	Amount of rainfall
SSH	Numerical	Sun shine hour
Humidity	Numerical	Average of humidity
Wind speed (m/s)	Numerical	Speed of wind

**Table 2** Visual representation of its datasets of FP Prophet

Parameter	Type	Explanation
Year	Numerical	Time
Month	Numerical	Time
M1	Numerical	Short-term scale
M3	Numerical	Short-term scale
M6	Numerical	Short-term scale
M9	Numerical	Short-term scale
M12	Numerical	Long-term scale

better and more drought value. Also, we have made another dataset that predicts the forecast of future drought.

From Table 2, we demonstrate short-term scale which indicates month-1, and month-3, month-6, and long-term scale indicates month-12 with their data types. We concatenate train and test data together for the analysis of our dataset. After that we get eight hundred and fifty rows of data, where most of the value not detected the drought. But previous-year data was detected according to the study.

### 3.3 Equations

The calculations of SPI explained by Giddings et al. [16] were followed in this study. The mean was first calculated

$$\text{Mean } X^- = \Sigma X N \quad (1)$$

After that standard deviation of 1.0 was calculated

$$S = \Sigma (X - X^-) N \quad (2)$$

The skewness was then calculated

$$\text{Skewness} = N(N - 1)(N - 2) \sum [X - X^- S]2 \quad (3)$$

The rainfall data was transformed by the log (In), and then the mean was calculated. The constant U, shape, and scale were also calculated

$$\text{Logmean } 0 = X^- \ln = \ln(X)N \quad (4)$$

$$U = \ln(X) - X^- \ln \quad (5)$$

$$\text{Shape} = \beta = 14U[1 + 4U3] \quad (6)$$

$$\text{Scale} = \alpha = X^- \beta \quad (7)$$

Here, the gamma distribution was performed with the shape and scale values

$$\text{Cumulative Gamma transform} = G(x) = 1\alpha\beta\Gamma\beta \int 0xx\beta - 1e - x\alpha dx \quad (8)$$

At the end, SPI is calculated with different formulas according to the magnitude of the gamma-transformed values

$$\begin{aligned} C_o &= 2.51555, \quad C_1 = 0.8028, \quad C_2 = 0.01032, \\ D_1 &= 1.43278, \quad D_2 = 0.18926 \text{ and } D_3 = 0.00130. \end{aligned} \quad (9)$$

For time series forecasting algorithm, we have used the FB Prophet which is the most accurate one for our study. The typical version is represented through the subsequent formula

$$y(t) = g(t) + s(t) + h(t) + e(t) \quad (10)$$

Here,  $y(t)$  represented cost of time collection at the time of  $t$ ,  $g(t)$  represents trend function of non-periodic changes which can be linear function or it can be logistic function,  $s(t)$  stands for changes according to yearly, monthly, and weekly that have seasonal component like yearly seasonal which using the Fourier series and weekly seasonal is using the dummy variables, and  $h(t)$  stands for the vacations and activities element where  $(t)$  is the mistake time period.

## 4 Results and Accuracies

From Table 3, Part 1 and Part 2, we demonstrate that short-term drought occurred in year 1981–2017 as considering moderate drought (M\_D), severe drought (Sev\_D), and extreme drought (Ext\_D). Also the drought index results are historical records of

drought periods (1–9 months) with some exceptions. Statistics to count the drought period a score is provided. The frequency of moderate droughts is higher in Dinajpur, part of Bangladesh and a drought-prone region; it is also a place where severe droughts frequently occur.

Table 4 shows that applying the dataset in the time series algorithm, we have observed that our value for RMSE is above 0.75, which indicates that the value we have got it and the model that have the good accuracy along with limited number of errors have been shown. RMSE measures the average of the squared variations among the forecasted values and the real values. It is sensitive to outliers and penalizes large errors more than MAE. It is the square root of the MSE and is expressed in the same units as the data. It provides a measure of the average magnitude of the error in the model predictions. Based on the RMSE value, it is shown that there will be no drought during the year of 2018 in the Dinajpur district. We also have shown the graph along with others values as well, when we have predicted the drought.

#### 4.1 *Interpreting SPI on Different Timescales*

The SPI for a given month is typically comparable with the average rate of precipitation for that month. In contrast, long-term precipitation records (over 30 years) are seen to be more accurate estimates of monthly precipitation for a specific region since they fit probability distributions, similar to other SPI durations. So that the median SPI for location and time is zero, it is then changed to a normal distribution. This means that half of the precipitation in the past has been below the median, and the other half has been above the median. A positive SPI value implies more precipitation than usual (i.e., wet weather), whereas a negative value suggests less precipitation than usual (i.e., dry conditions).

Figure 3 shows that the SPI value of 1-months. In this case, the 1-month SPI is a short-term value and could be significant for linking soil moisture and plant stress during the growth season. Red symbol is showing the occurrence of drought for 1-month scale or indicates as dry.

Figure 4 shows the SPI value of 3-month. In this case, the 3-month SPI offers a comparison between the total amount of precipitation for a certain 3-month period and all years in the historical record or to put it another way, and the late-February three-month SPI is the total of the precipitation from December through January and February for the specific year and the total precipitation from December through February for all years. Comparing the 3-month SPI gives projections of seasonal precipitation as well as short- and medium-term humidity levels. It is crucial to contrast the 3-month SPI with longer time periods. Prolonged droughts, which are only visible on longer time scales, can produce rather typical 3-month intervals.

Figure 5 represents the SPI value of 6-month. Here, the 6-month SPI analyzes the current period's precipitation with the same 6-month duration in historical records. The 6-month SPI exhibits medium-term precipitation trends and is thought to be more responsive to such extreme situations. A 6-month SPI is highly useful for displaying

**Table 3** Drought frequency in Dinajpur for numerous quick month lengths with SPI*Part 1*

Year	Month-1			Month-3			Month-6			Month-9		
	M_D	Sev_D	Ext_D									
1981	0	1	1	2	0	0	0	0	0	0	0	0
1982	1	0	0	3	0	0	8	0	0	5	1	0
1983	2	0	0	0	0	0	0	1	0	3	1	0
1984	0	0	0	0	0	0	0	0	0	0	0	0
1985	0	0	0	2	0	0	2	0	0	0	0	0
1986	0	0	0	1	0	0	1	0	0	0	0	0
1987	0	0	0	0	0	0	0	0	0	0	0	0
1988	0	0	0	0	0	0	0	0	0	0	0	0
1989	1	0	0	0	0	0	0	0	0	0	0	0
1990	0	0	0	0	0	0	0	0	0	0	0	0
1991	1	0	0	0	0	0	0	0	0	0	0	0
1992	2	0	2	1	0	3	0	2	3	0	0	2
1993	1	0	0	2	0	0	0	2	0	0	0	0
1994	1	1	0	1	1	2	0	2	3	1	1	4
1995	1	1	0	1	1	1	1	2	1	0	3	3
1996	1	0	0	1	1	0	1	1	0	0	0	0
1997	1	1	0	0	0	0	0	0	0	0	0	0
1998	1	0	0	0	0	0	0	0	0	0	0	0
1999	0	0	0	2	0	0	0	0	0	0	0	0
2000	2	0	0	3	0	0	1	1	1	2	0	0
2001	1	1	0	2	0	1	0	3	1	1	2	2
2002	0	0	0	0	0	0	0	0	0	0	0	0
2003	0	0	0	0	0	0	0	0	0	0	0	0
2004	1	0	0	0	0	0	0	0	0	0	0	0

*Part 2*

Year	Month-1			Month-3			Month-6			Month-9		
	M_D	Sev_D	Ext_D									
2005	0	0	0	0	0	1	0	0	0	0	0	0
2006	3	0	0	5	0	0	2	4	0	2	4	0
2007	0	0	0	2	0	0	1	0	0	1	3	0
2008	1	0	0	2	0	0	2	0	0	2	0	0
2009	1	0	0	1	0	0	2	0	0	0	0	0
2010	0	1	0	4	1	0	1	0	0	0	0	0

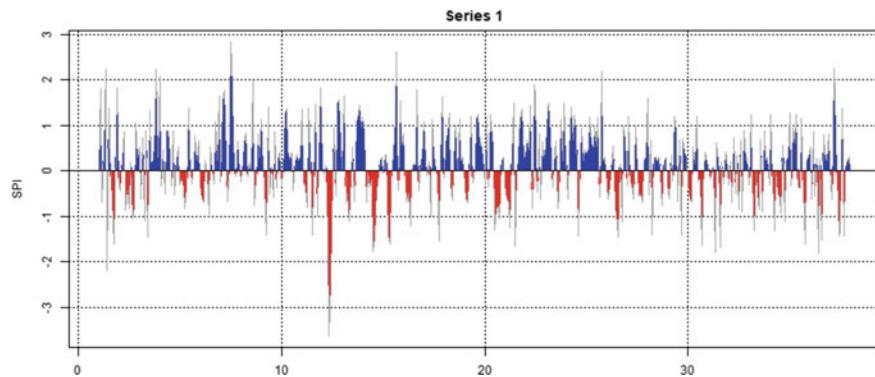
(continued)

**Table 3** (continued)

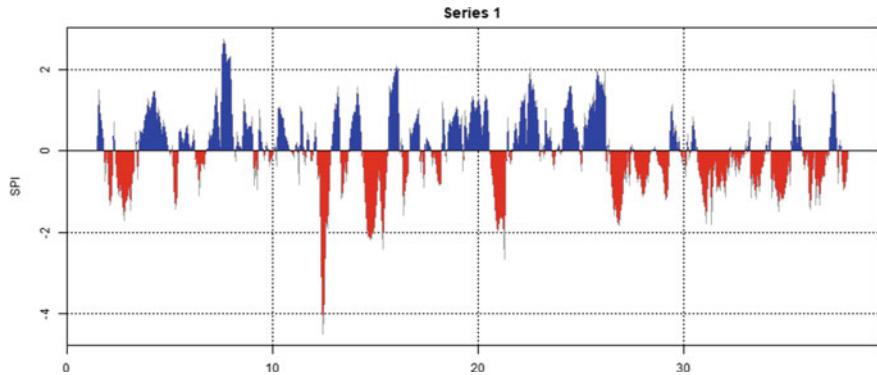
Year	Month-1			Month-3			Month-6			Month-9		
	M_D	Sev_D	Ext_D									
2011	0	2	0	1	1	0	1	2	0	6	0	1
2012	0	0	0	0	0	0	1	0	0	2	0	0
2013	0	0	0	1	0	0	2	0	0	1	0	0
2014	2	0	0	2	0	0	6	0	0	6	0	0
2015	0	0	0	1	1	0	0	0	0	2	0	0
2016	0	0	0	1	1	0	4	0	0	2	1	0
2017	3	0	0	1	1	0	0	0	0	0	0	0
2018	27	11	3	42	10	8	36	20	9	36	16	12

**Table 4** Forecast results for FP Prophet model

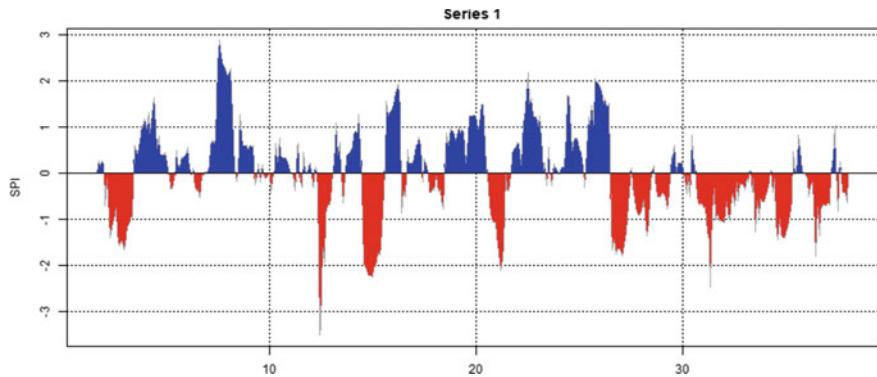
MSE (mean squared error)	1.380152
RMSE (root mean square error)	1.171201
MAE (mean absolute error)	0.900316
MAPE (mean absolute percentage error)	1.909277
MAPE (median absolute percentage error)	1.072456
Coverage (accuracy)	0.75

**Fig. 3** SPI-1

precipitation throughout the duration of the year. A connection between irregular flows and resource levels and information from the 6-month SPI may also begin to show up.



**Fig. 4** SPI-3

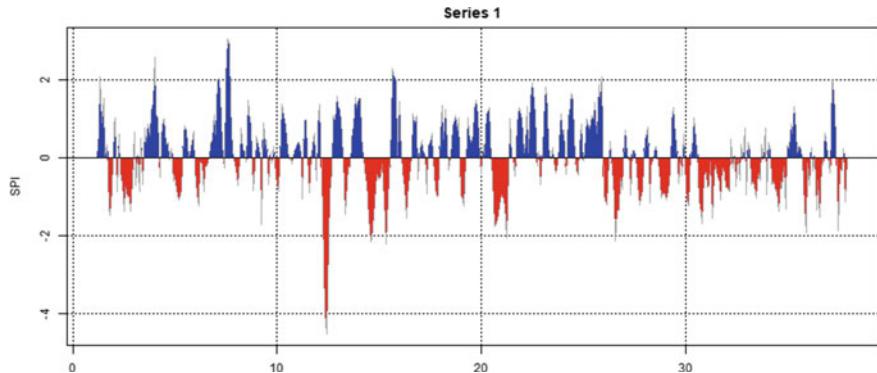


**Fig. 5** SPI-6

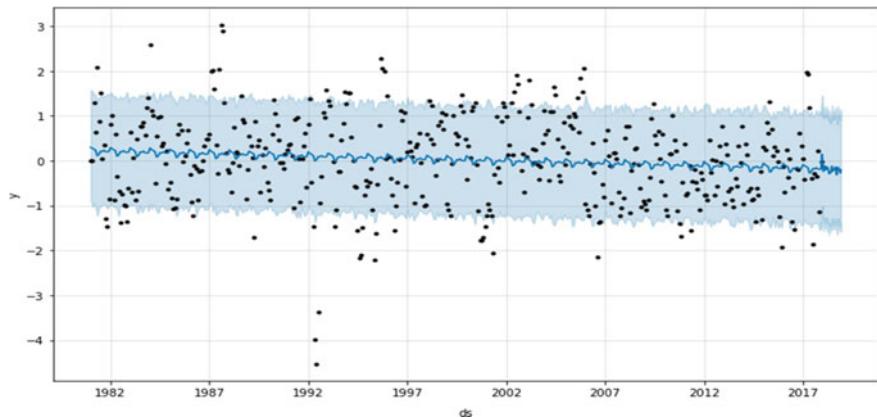
In Fig. 6, we demonstrate the SPI value of 9-month. The 9-month SPI shows an intermediate precipitation pattern. It usually takes more than one season for a drought to occur. SPI values below  $-1.5$  on these timescales typically indicate that fairly significant impacts are occurring in agriculture and may manifest in other sectors as well.

#### 4.2 FB Prophet Prediction

According to above Fig. 7, it shows that time series algorithm predicted value for next 365 days or after 2017 based on the dataset. We have used here previous datasets, and we found probable value of next years. The predicted projection plot displays the forecasted values and uncertainty intervals for a time series. This plot provides an overall view of the performance of the forecasting model and can be used to



**Fig. 6** SPI-9

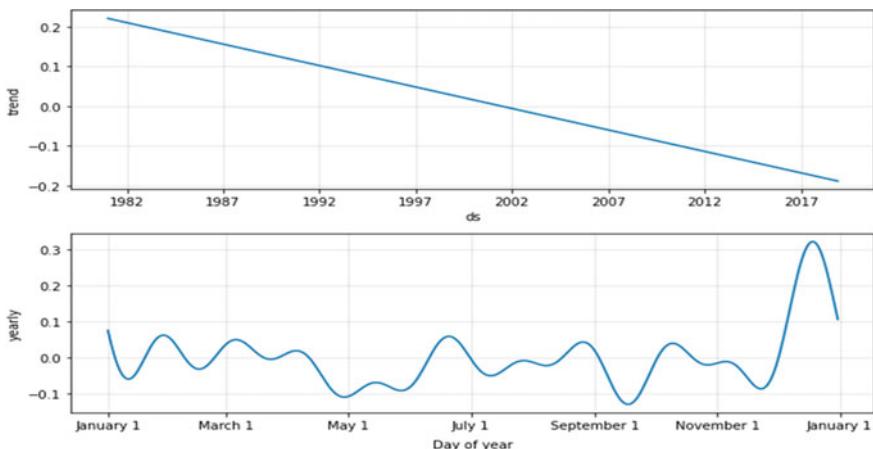


**Fig. 7** Predicted projection

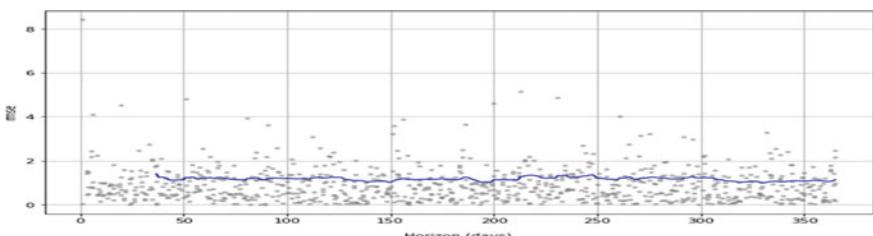
understand how well the model is capturing the underlying trends and patterns in the data. In a predicted projection plot, the x-axis represents the time index, and the y-axis represents the target variable being forecasted. The observed data is typically shown as a solid line, while the forecasted values are shown as a dotted line. The uncertainty intervals are shown as shaded regions around the forecasted line, and they represent the range of values within which the actual values are expected to lie with a certain degree of confidence. The predicted projection plot can be used to identify trends and patterns in the data, assess the performance of the model, and identify areas for improvement. If the observed data and the forecasted line are well aligned, it suggests that the model is capturing the underlying trends and patterns in the data. If the uncertainty intervals are narrow, it suggests that the model is confident in its forecasts. On the other hand, if these uncertainty intervals are wide or the observed data and the forecasted line diverge significantly, it may indicate that the model is not capturing the underlying patterns in the data or that the data is highly volatile.

Figure 8 shows that seasonality plot displays the observed data and the forecasted seasonality, along with the uncertainty intervals. Prophet also provides the ability to customize these plots and add additional information to them, such as markers for events or annotations. These plots can be used to understand the performance of the model, diagnose issues, and gain insights into the data. According to our survey, we have also observed the yearly and monthly trend of predicated value. This plot provides an understanding of how well the model is capturing the seasonal patterns in the data. The plot also showing the year-wise value of the drought is decreasing in a certain amount, so that the drought occurring rate is increasing. It also showing that the rate of drought value is most in January, but the low rate in the September which is less than  $-1$ . Day by day, the label of the rain is decreasing, and the drought level is increasing.

Figure 9 shows that metrics are numerical values used to evaluate the performance of a model, such as the accuracy of its predictions. The accuracy of the metrics depends on several factors, including the quality of the data, the choice of model, the assumptions made about the data, and the nature of the forecasting problem. According to the RMSE value, it is predicted that there will not be a drought in the



**Fig. 8** Initialize each component (tends, yearly)



**Fig. 9** Prediction with RMSE without error value

Dinajpur district in 2018. Because according to the graph, the value of the RMSE is not laying under zero value and matching with Table 5 values, which confirm that there will not be a drought in the Dinajpur district in 2018. Different metrics may provide different insights into the performance of a model, and the choice of metric should depend on the specific requirements of the forecasting problem. When we forecast the drought, we also displayed the graph along with other values.

## 5 Conclusion and Future Endeavors

Our main focus in this study was drought circumstance over Bangladesh which analyzed the use of standardized precipitation index (SPI) and Shiny from the rainfall statistics of the Bangladesh Meteorological Department in Dinajpur District at some stage from 1981 to 2017. In this research we observed that, SPI calculation of moderate drought is higher than the other droughts. This indicates that drought is ordinary again and again in Bangladesh, performed extra, or much less in Dinajpur. Based on the research, we can also observe from the beginning of the year using the times series forecasting algorithm, which gives the most accurate forecast result based on drought prediction. Due to the above process, we can easily detect the drought from the beginning of the year and take necessary steps according to that. So our government can take necessary steps against drought to mitigate and minimize it for future.

In the future, we will strive to collaborate with more robust datasets from various reputed stations in Bangladesh. Time series forecasting algorithm gives better accuracy to identify drought in a particular location within a confined period. We will observe other nearby locations for detecting small element drought situation like sub-district and also implementing the LSTM and ARIMA for better comparison and predict more than one-year data. For that reason, we are able to expect sub-district smart drought in keeping with the SPI month.

**Table 5** Drought years (dy), drought-affected areas (dar), and ancient information of drought detection with the aid of using nearby evaluation use of SPI in Bangladesh from 1981 to 2017 and predicted 2018

<i>Part 1</i>		
Year	Area	SPI
1981	Dinajpur	Detected
1982	Dinajpur	Detected
1983	Dinajpur	Detected
1984	Dinajpur	Not detected
1985	Dinajpur	Detected
1986	Dinajpur	Detected
1987	Dinajpur	Not detected
1988	Dinajpur	Not detected
1989	Dinajpur	Not detected
1990	Dinajpur	Detected
1991	Dinajpur	Detected
1992	Dinajpur	Detected
1993	Dinajpur	Detected
1994	Dinajpur	Detected
1995	Dinajpur	Detected
1996	Dinajpur	Detected
1997	Dinajpur	Detected
1998	Dinajpur	Detected
1999	Dinajpur	Detected
2000	Dinajpur	Detected
<i>Part 2</i>		
Year	Area	SPI
2001	Dinajpur	Detected
2002	Dinajpur	Detected
2003	Dinajpur	Not detected
2004	Dinajpur	Detected
2005	Dinajpur	Not detected
2006	Dinajpur	Detected
2007	Dinajpur	Detected
2008	Dinajpur	Detected
2009	Dinajpur	Detected
2010	Dinajpur	Detected
2011	Dinajpur	Detected
2012	Dinajpur	Detected
2013	Dinajpur	Detected
2014	Dinajpur	Detected
2015	Dinajpur	Detected

(continued)

**Table 5** (continued)

Part 2		
2016	Dinajpur	Detected
2017	Dinajpur	Detected
2018	Dinajpur	Not detected

## References

1. Achite AM, Elshaboury N, Jehanzaib M, Vishwakarma DK, Pham QB, Anh DT, Abdelkader EM, Elbeltagi A. Performance of machine learning techniques for meteorological drought forecasting in the Wadi Mina Basin
2. Pal JS, Small EE, Eltahir EAB (2000) Simulation of regional scale water and energy budgets: representation of subgrid cloud and precipitation processes within RegCM. *J Geophys Res* 105(D24):29579–29594. <https://doi.org/10.1029/2000JD900415>
3. Rafiuddin M, Dash BK, Khanam F (2011) Diagnosis of drought in Bangladesh using standardized precipitation index. In: International conference on environment science and engineering IPCBEE. IACSIT Press, Singapore, vol 8
4. Shahid S, Behrawan H (2008) Drought risk assessment in the western part of Bangladesh. *Nat Hazards* 46:391–413. <https://doi.org/10.1007/s11069-007-9191-5>
5. Giorgi F, Marinucci RM (1996) An investigation of the sensitivity of simulated precipitation to the model resolution and its implications for climate studies. *Mon Wea Rev* 124:148–166. Young M (1989) The technical writer's handbook. University Science, Mill Valley
6. Lee J, Hwang Y, Kim T (2020) Forecasting drought indices using machine learning algorithm paper presented at 2020. In: ASEE virtual annual conference content access, virtual online. <https://doi.org/10.18260/1-2-34680>
7. Brust C, Kimball JS, Maneta MP, Jencso K, Reichle RH. Drought cast: a machine learning forecast of the United States drought monitor
8. Duan S (2022) AutoML-based drought forecast with meteorological variables. In: Atmospheric Science Graduate Group University of California, Davis Davis, CA 95616, [shiduan@ucdavis.edu](mailto:shiduan@ucdavis.edu), 15 July 2022
9. Prodhan DFA, Zhang J, Yao F, Shi L, Pangali Sharma TP, Zhang D, Cao D, Zheng M, Ahmed N, Mohana HP. Deep learning for monitoring agricultural drought in SouthAsia using remote sensing
10. Khan N, Sachindra DA, Shahid S, Ahmed K, Shiru MS, Nawaz N (2020) Prediction of droughts over Pakistan using machine learning algorithms. *Adv Water Resour* 139:103562
11. Ullah K, Wang Y, Fang Z, Wang L, Rahman M (2022) Multi Hazard susceptibility mapping based on convolutional neural networks. *Geosci Front* 13(5):101425
12. Vodounon RBW, Soude H, Mamadou O (2022) Drought forecasting: a bibliometric analysis and future research directions. *J Environ Protect* 13:972–990
13. Mokhtar A, Jalali M, He H, Al-Ansari N, Elbeltagi A, Alsafadi K, Abdo HG, Sammen SSH, Gyasi-Agyei Y, Rodrigo-Comino J (2021) Estimation of SPEI meteorological drought using machine learning algorithms. *IEEE Access* 9
14. Nandgude N, Singh TP, Nandgude S, Tiwari M (2023) Drought prediction: a comprehensive review of different drought prediction models and adopted technologies. *Sustainability* 15:11684
15. Kan J-C, Ferreira CSS, Destouni G, Haozhi P, Passos MV, Barquet K, Kalantari Z (2023) Predicting agricultural drought indicators: ML approaches across wide-ranging climate and land use conditions. *Ecol Ind* 154:110524
16. Giddings L, Soto M, Rutherford BM, Maarouf A, (2005) ‘Standardized precipitation index zones for Mexico’, *Atmosfera* 18(1), 33–56

# A Survey on Various Aspects of Recommendation System Based on Sentiment Analysis



Rohit Mittal, Sumit Kumar, Vishal Shrivastava, Vibhakar Pathak,  
and G. L. Saini

**Abstract** Many industries, including e-commerce, media, finance, and utilities, have embraced recommender systems. To maximize customer happiness, this type of technology uses a vast quantity of data. These recommendations assist customers in selecting items, while companies can enhance product use. When it comes to analyzing social data, sentiment analysis may be used to acquire a better knowledge of users' thoughts and feelings, which is useful for enhancing the dependability of recommendation systems. However, this data may also be utilized to supplement user ratings of items. According to some, sentiment analysis (SA) of articles that may be found in online news sources and blogs or even in the recommender systems themselves can provide better recommendations to users. Research trends that connect sophisticated technological components of recommendation systems utilized in many service domains with the commercial aspects of these services are reviewed in this article. We must first conduct an accurate evaluation of recommendations models for recommendation systems (RS) using data mining and application service research. Deep learning architectures for breast cancer detection are the topic of this review. The following is a list of current machine learning-based technologies that will be discussed in this survey. Research into recommendation systems is made possible by this study's examination of the numerous technologies and service trends to which recommendation systems may be applied, which gives a complete overview of the area.

**Keywords** Recommendation system · Content-based filtering · Collaborative filtering · Recommendation technique

---

R. Mittal · G. L. Saini (✉)  
Manipal University Jaipur, Dahmi Kalan, Rajasthan, India  
e-mail: [glsaini86@gmail.com](mailto:glsaini86@gmail.com)

S. Kumar · V. Shrivastava · V. Pathak  
Arya College of Engineering and IT, Jaipur, India  
e-mail: [vibhakar@aryacollge.in](mailto:vibhakar@aryacollge.in)

## 1 Introduction

The widespread use of Web 2.0 has created an atmosphere where customers may express themselves, be creative, communicate, and share. Many consumers use online ordering platforms (like Dianping and TripAdvisor) to share their thoughts about restaurants and their feelings about them. It's a great way to meet new people while doing something fun. A large number of restaurant evaluations written by customers themselves allow customers to communicate their wants and requirements while also supporting businesses in offering timely and personalized service [1, 2]. With an average of nine reviews per person each year, two-thirds of consumers have taken the time to evaluate local businesses [3]. Restaurant clients rely extensively on customer evaluations to determine the quality of service before spending money since goods and services are intangible and complicated. [4]. In addition to expressing the emotional requirements of customers, restaurant reviews serve as a significant source of information to which consumers may turn for information [5]. Consumers prefer to seek a high number of restaurant reviews from other users during the pre-consumer information search phase to lessen the perceived uncertainty and perceived risk created by information asymmetry [6].

The recommendation system area has been of significant relevance in academia, business, and industry since its inception more than a few decades ago. Amazon, Pandora, Netflix, TripAdvisor, Yelp, Facebook, and articles are just a few of the places where they've been used articles (TED). As e-commerce websites have evolved, it has become clear that RSs can be a valuable tool for helping customers find products that are a good fit for their wants and requirements. As an example, 35% of Amazon sales/revenues came from user-recommended goods in 2015.

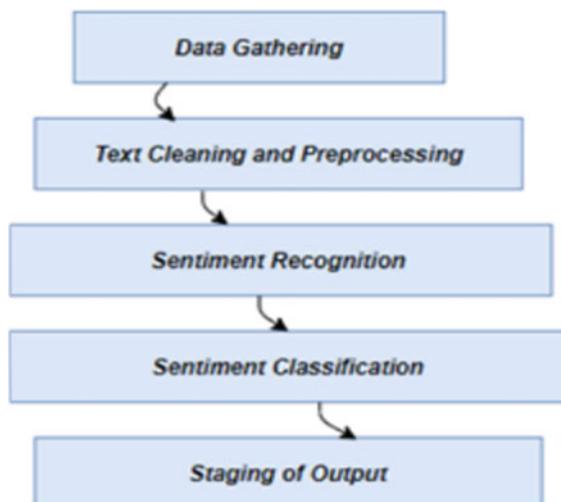
The three most common methods of making recommendations are collaborative-based, content-based, and hybridized versions of all three. According to what the user liked, purchased, or watched, content-based (CB) method mines relevant recommendations for the user [7]. A suggestion is generated for a user using the collaborative filtering (CF) technique, which compares his prior preferences and interests to those of other users [8]. For those who still want additional options, there's the hybrid method, which combines recommendations from several different sources into a single recommendation system [9]. Nevertheless, these conventional RS techniques rely on the recommendation process being based on a single-criteria rating (overall rating). For a suggestion, a single-criterion rating is insufficient since the overall ratings cannot represent the fine-grained understanding underlying the user's behavior. Study on how to improve the RS's performance has become a broad research issue as a result of this. User suggestions are provided via recommender systems, which are software tools and techniques [10]. Recommender systems aid customers in making judgments about the products or things they want to buy. To produce suggestions, recommender systems process information from a variety of sources, including data that is being actively collected. Depending on the recommender system, different types of data were employed to process the information [11]. As a result of these

structures, we can make suggestions regarding products or items based on information that is readily available through online social networks (OSN). The cloud platform provides an automatic recommender system that may propose products or things depending on the user's inquiries; this is done through the cloud platform. In the cloud, users may share their thoughts and sentiments about things and items. For the recommender system, online social networks like Facebook or Twitter will be an excellent cloud platform.

On social media, customers are increasingly discussing their experiences with one another. Many customers rely on their purchasing decisions on what other people think of service or product. As a result of this phenomenon, the number of online viewpoints has grown rapidly (that is, user reviews). It is the opinion of the customer that is expressed in each review, whether it be a purchase, a movie, or a lodging reservation. Consumers and companies alike value product reviews like this. However, despite the advantages of these assessments, it is extremely difficult to extract relevant information from them due to their massive scale and unique qualities [12]. The reviews' qualities make it harder for robots to understand written natural language compared to other structured data sources; hence, most RRs do not use them in creating suggestions [13]. In addition to encouraging customers to consider the viewpoints of others, sentiment analysis aids in the selection of purchases based on the opinions of other customers. To meet the needs of advertising and product benchmarking in the industrial company, assessment mining may also be used to highlight product improvement. SA procedure is depicted in Fig. 1.

The goal of SA might be in the form of speech, text, pictures, or any other form of communication. Because restaurant reviews are often delivered as text, the majority of articles on sentiment analysis rely on text-based SA [14]. Consumers often establish a broad perception of a restaurant during the pre-purchase information-seeking stage, and the massive volume of restaurant review material surpasses consumers'

**Fig. 1** Sentiment analysis process



capacity to comprehend it, and reading fewer reviews improves the probability of forming misperceptions [15]. The platform must be capable of processing information promptly to immediately detect the emotional information present in restaurant reviews [16].

## 2 Background

### 2.1 *Sentiment Analysis (SA)*

In addition to encouraging customers to consider the viewpoints of others, sentiment analysis aids in the selection of purchases based on the opinions of other customers. To meet the needs of advertising and product benchmarking in the industrial company, assessment mining may also be used to highlight product improvement. The sentiment analysis procedure is depicted in Fig. 1. Lexicon-based techniques were first to be utilized for SA. There are two schools of thought: lexical and corpus-based [17]. To classify emotions in the former case, a dictionary of words, like SentiWordNet or WordNet, is utilized. But SA based on corpus-based SA does not depend upon predetermined lexicon nonetheless on a statistical analysis of contents in documents collection, like k-NN [18], CRF [19], and HMM [20], among others.

Machine learning-based techniques [21] traditional methods and deep learning are the two main approaches to solving sentiment analysis difficulties. Naive Bayes [22] maximum entropy [23, 24] and Support Vector Machines (SVMs) [25] are examples of traditional machine learning methods. Lexical and sentiment characteristics, sections of the speech, or adjectives and adverbs are all examples of input to these algorithms. How accurate these systems can be is based on the features that are selected. More effective than more conventional methods may be achieved using deep learning. SA may make usage of different types of deep learning (DL) models, like RNN, CNN, and DNN. This section discusses categorization models that handle issues at the document, phrase, or aspect levels. The hybrid approaches [26] ML and lexicon-based techniques can be utilized together. For the most part, these tactics rely heavily on the use of emotive lexicons.

### 2.2 *Recommender System*

Product or service suggestions are provided by a recommendation system to help consumers make better decisions as Internet information continues to grow. E-business, E-government, and e-shopping or e-commerce are only a few of the many systems that have been developed and put into use in the three primary areas of government and business, as well as education [28]. Recommendation systems are

commonly used in e-commerce to help buyers pick from a variety of items. Using a filtering strategy, systems for delivering tailored options have improved [29].

Content-based, collaborative filtering (CF), and hybrid recommender systems (HRS) are the three most prevalent methodologies for recommender systems. Based on the sort of social media data that is being analyzed, these strategies might differ. An analysis of common recommender systems by Lu and colleagues [28] efficiently reveals what is needed in the area. Additionally, our activity actively encourages and supports academics and practitioners to encourage widespread use and use of recommender systems in a variety of industries and contexts.

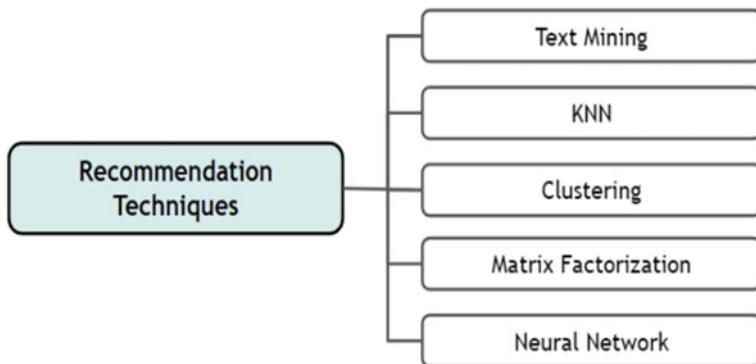
**Content-Based Recommender Systems:** Users' profiles and item attributes are used in content-based techniques. For instance, product characteristics can be utilized to generate user profiles from the content of products accessible over the Internet by the user. Using content-based similarity measurements between catalogue items and those that users have consumed, accessed, or rated favorably, recommender systems filter items. As a result, a user is shown things that are comparable to those that they have previously found interesting. A quantitative study of an item's information may be used to derive its utility for a particular user.

**Collaborative Filtering-Based Recommender Systems:** Cooperative filtering is a method for removing goods that people might appreciate based on the opinions of others who have used the same product. It does this by looking through a big number of individuals and identifying a smaller group of people whose likes are similar to the users. It analyzes what the user likes and creates a ranked list of suggestions based on that. To utilize recommender algorithms, we require data that includes a collection of goods and a set of users. With this data, the matrix is comprised of the responses supplied by a group of users to certain objects within a collection. A user's ratings would appear in each row, and an item's ratings would appear in each column.

**Hybrid Recommender Systems:** Any data that may be gleaned or inferred from online platforms, social media, or additional sources can be used in a hybrid approach. The typical issues in recommender systems can be addressed by a generic consolidative model based on a combination of individual deployment and accumulation of rankings and forecasts. Collaborative filtering, for example, has sparseness, scalability, and cold-start difficulties [29, 30]. Each recommendation technique has advantages and drawbacks. When we have a lot of data, we have a sparseness problem. When a person or item is introduced to the system, a cold-start issue develops because of a lack of rating data. Solving these issues may be made easier by combining sentiment analysis and recommendation approaches.

### 3 Recommendation Techniques

Data mining is a method for finding patterns and connections in massive datasets using statistical analysis. To classify client or visitor clickstream data matching the client or visitor group, it analyzes the item information, provides recommendations to the user, and also builds comparable user groups among users. For fulfilling



**Fig. 2** Technology mainly used in recommendation system

the demands of individual users, it can also offer personalized browsing alternatives. Recommendations based on various data mining approaches can be generated. Figure 2 provides a visual representation of the strategies that will be discussed in this section.

### Text Mining

Data may be mined for important text information by extracting text-related information. The semantically relevant information has been retrieved from the accompanying text thanks to current advances in natural language processing technology. A limitation of comprehending semantics exists when natural language processing (NLP) is utilized in various text analysis methods because of the inclination to assess texts based on the frequency of words. As a result, the ontology, which specifies the common vocabulary of things and organizes meaning by establishing conceptual schema of text-domain, began to be employed to correctly comprehend the meaning of the text.

### K-Nearest Neighbor (KNN)

To categorize a dataset, the K-nearest neighbor (KNN) method sorts test and training tuples based on their K-nearest neighbors. KNN uses distance-based weighting to compare the similarity between each piece of data to classify datasets. Pearson correlation, Euclidean distance, and cosine similarity are the most often used methods for comparing similarity. Using the KNN algorithm, a recommendation system may classify a user's search habits and forecast what products the user would like in the future. It is possible to categorize objects that are similar to user likes based on patterns in user activity data, like click stream data and web server logs, and then utilize the findings to propose appropriate goods.

### Clustering

It is common practice in recommendation systems to employ clustering as a method of classifying data into discrete categories or clusters [27]. K-means clustering is

the most often used clustering approach in the recommendation system. A technique known as K-means clustering gathers data into groups centered on the mean once a predetermined number of K clusters has been selected. Following the computation of all of the data in RS, the data is allocated to the nearest cluster, and calculation is repeated in a sequence of computing cluster centers. It is, nevertheless, sensitive to the scalability problem, in which the computation performance reduces as the number of users and objects rises when a recommendation system is being serviced by K-means clustering.

## 4 Literature Review

For SA of online ordering platform reviews, we used an attention mechanism and a Bi-GRU. Online restaurant reviews and sentiment analysis approaches are discussed in this section.

### 4.1 *Online Restaurant Reviews*

When it comes to choosing a restaurant, customers often look to customer evaluations in addition to further information offered by merchants, like expert advice, restaurant descriptions, and recommendations based on their preferences [31]. To build a general impression of a store, customers who read restaurant reviews would draw on their prior experiences to form an opinion of the business, which in turn influences their purchasing decisions. As a result of checking for Internet restaurant evaluations, many consumers choose to dine at a well-known establishment.

Table 1 lists some of the most significant research articles in the area of online restaurant reviews and consumer psychology and behavior.

In online evaluations also other types of computer-mediated communication, consumers convey their emotions. Emotional data from Internet restaurant evaluations was gleaned by some academics to give practical advice. Researchers utilized the DL method to investigate aspect restaurant sentiment during the COVID-19 pandemic period also found that the DL model performed better overall than machine learning algorithms. Restaurant review sentiment is classified using Naive Bayes, a method that helps marketers understand the preferences and characteristics of their customers. SA of restaurant reviews has been investigated from a methodological standpoint by certain researchers [37, 38]. For SA, authors utilized the word co-occurrence technique to determine how many times certain words occur together in a sentence. The authors then used this information to determine which sentences had the highest implicit feature scores, and findings demonstrated that this threshold-based approach performed well [27]. The emotional intensity of online reviews was assessed using a text mining algorithm, and an experimental study indicated that pleasant emotions had a negative influence on reviews, while negative emotions had

**Table 1** Literature about online restaurant reviews

Study	Theme	Methods	Conclusion
[31]	Customers' eating experiences vary greatly in their level of satisfaction	Quantitative data on the sentiments expressed by users in their comments	Cross-cultural social commerce platforms may see considerable variances in user behavior
[32]	Investigate the narratives people tell about their happy and negative emotions on the Internet	Text mining	Trauma narratives are more likely to be found in negative reviews, but lengthy narratives in good reviews are more likely to highlight the reviewer's linguistic capital
[33]	The restaurant consumer motivation and happiness of customers from diverse cultural backgrounds may be studied and compared	Probabilistic theme model	In contrast, Chinese visitors are less likely to disparage restaurants and are more captivated by the cuisine on offer, American visitors, on the other hand, are more inclined to seek enjoyment and are less afflicted by crowds
[34]	Does the supposed practicality of ratings shown by "likes" depend on the perceived normative and informational qualities of online restaurant reviews?	Content analysis	Leads can benefit greatly from heuristic filtering
[35]	Online review data may be used to gauge customer mood and emotional responses to meals	Empirical and dictionary-based sentiment analysis	More emotive language was used to describe restaurant service than food in customer evaluations, with more people emphasizing the good than the bad
[36]	How the length of a trial influences the consistency of review evaluations	Empirical analysis	The type of review equipment and the empirical value of the data has a significant moderating influence on the link between time and review consistency

a beneficial effect, and expressing furious feelings was more effective than good emotions [39]. SA of online restaurant reviews was performed by Krishna et al. using ML techniques, and SVM produced the best results when applied to a specific data set.

## 4.2 *Sentiment Analysis Method*

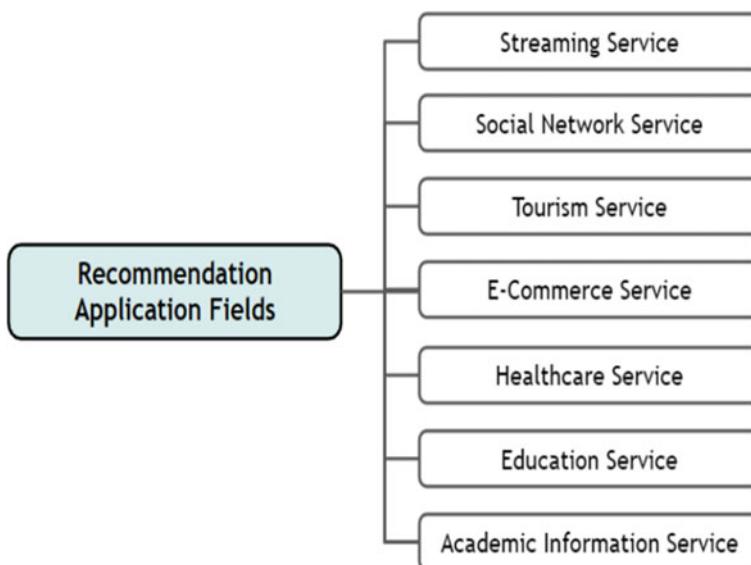
As a computer investigation of people's demands, attitudes, and feelings about a thing, SA is also called opinion mining. Findings of SA may be used in a variety of domains, like topic monitoring, online sentiment opinion analysis, word-of-mouth evaluation of enormous items, etc. Selection of features from subjective texts is a critical part of sentiment analysis, and doing so well may boost the accuracy of sentiment analysis greatly. Researchers have spent a great deal of time looking at features in an attempt to discover a good way to pick features. According to researchers [36], the Boolean weighting approach that they utilized to determine feature weights resulted in greater accuracy than the feature characterization methods they picked. A more domain-specific approach to sentiment analysis necessitates domain-specific information to increase the system's performance; in this way, product feature selection may be viewed as a process of identifying domain-specific named entities. Sentiment analysis is less effective when it is applied outside of a specific context, and existing studies on feature selection are limited.

Sentiment dictionaries and machine learning approaches have been employed in some research to assess restaurant reviews; however, the data processing effort and the domain are less transferrable. As a result, deep learning-based sentiment analysis approaches are becoming more popular because they offer automated feature extraction, richer representation performance, and higher performance. Traditional methods of sentiment classification lose both temporal and positional information, so Abdi et al. presented DL-based method (called RNSA) to classify user opinions expressed in reviews. This approach outperforms the traditional methods in terms of sentiment classification at the sentence level. Al-Smad utilized Long Short-Term Memory (LSTM) to analyze reviews of Arabian hotels in two ways: First, by integrating Bi-LSTM and conditional random fields for the formulation of opinion requirements classification, and second, by employing LSTM for SA, which both surpassed the prior baseline research, it was discovered that both were superior to the prior baseline study. Sentiment dictionaries and classical machine learning are commonly utilized in the field of SA [40]. Because the performance of the model is largely influenced by the feature selection technique and parameter adjustment, these strategies are ineffective. CNN, RNN, LSTM, also other network topologies are included in DL. Neural networks are utilized in DL-based SA models because they can learn to extract complicated characteristics from data with minimum external contributions. SA depends upon DL is more generalizable also has superior performance in terms of feature extraction and nonlinear fitting than sentiment analysis based on machine learning.

## 5 Applications

The recommendation system has been employed in a wide range of service industries. These models and technology for recommendation systems outlined above will be examined in this research to see how they may be applied to a specific service field. Streaming services, social network services, e-commerce services, tourism services, education services, healthcare services, and academic information services were all included in the recommendation system's scope of use. Recommendation systems with rising user or commercial value and services that emerge often when "recommendation system" is searched in the Google Scholar search engine are broken down into seven primary categories. As seen in Fig. 3, this section's recommendation system relies heavily on several services.

Due to the growing popularity of Internet-connected smartphones, it is now possible to provide mobile users with personalized and context-sensitive suggestions, and hence, more mobile RSs are required. A more difficult problem arises when dealing with diverse, noisy, and time- and space-dependent mobile data. In the realm of RS, the more mobile-based analysis might have a substantial impact.



**Fig. 3** Recommendation systems used in this study may be found here

## 6 Conclusion

The proliferation of the Internet, mobile devices, and Social Networking Sites (SNS) has led to an increase in the number of online and application services. Thus, it is imperative to build a wide range of RSs that can assist consumers in quickly receiving and making selections in the face of an ever-increasing amount of item information. As a result, wearable gadgets and click stream data combined with real-time recommendation systems often lead to improved outcomes. A healthcare recommendation system's outputs, such as a diagnosis and treatment plan, have lower affinity than those depending upon clinical data. Real-time data, on the other hand, assures a more relevant outcome by reflecting patients' present state and can provide prompt advice for consultation services and urgent remedies.

Study on RSs was examined from a macro-viewpoint, including instances of service applications and interconnection between RS-related study and business of application service. For academics interested in recommendation systems, this was meant to provide a general perspective. This work will serve as a foundation for future research into the creation of recommendation systems tailored to definite requirements of businesses operating in the application service sector.

## References

1. Anaya-Sánchez R, Molinillo S, Aguilar-Illescas R, Liébana-Cabanillas F (2019) Improving travellers' trust in restaurant review sites. *Tour Rev* 74:830–840
2. Chen Y, Xie J (2008) Online consumer review: Word-of-mouth as a new element of marketing communication mix. *Manag Sci* 54:477–491
3. Local Consumer Review Survey. Available online: <https://www.brightlocal.com/research/local-consumer-review-survey/>. Accessed on 1 Jan 2019
4. Yang S-B, Hlee S, Lee J, Koo C (2017) An empirical examination of online restaurant reviews on Yelp.com: a dual coding theory perspective. *Int J Contemp Hosp Manag* 29:817–839
5. Marine-Roig E, Clave SAA (2015) A method for analysing large-scale UGC data for tourism: application to the case of Catalonia. In: *Information and communication technologies in tourism 2015*. Springer, Berlin, pp 3–17
6. Hong H, Xu D, Wang GA, Fan W (2017) Understanding the determinants of online review helpfulness: A meta-analytic investigation. *Decis Support Syst* 102:1–11
7. Pazzani MJ, Billsus D (2007) Content-based recommendation systems. *Dlm. (pnyt.). In: The adaptive web*. Springer, pp 325–341
8. Acar S, Zhang D, Simoff S, Debenham J (2007) Informed recommender: basing recommendations on consumer product reviews. *IEEE Intell Syst* 22(3):39–47
9. Danilova V, Ponomarev A (2016) Hybrid recommender systems: the review of state-of-the-art research and applications. In: *Proceeding of the 20th conference of FRUCT Association*
10. Esmaeili L, Mardani S, Hashemi Golpayegani SA, Zanganeh Madar Z (2020) A novel tourism recommender system in the context of social commerce. *Exp Syst Appl* 149(1)
11. Hedge SB, Satyappanavar S, Setty S (2018) Sentiment based food classification for restaurant business. In: *International conference on advances in computing, communications and informatics*, IEEE, Bangalore, India, pp 1455–1462
12. Chen L, Chen G, Wang F (2015) Recommender systems based on user reviews: the state of the art. *User Model User-Adap Inter* 25(2):99–154

13. Musat C-C, Liang Y, Faltings B (2013) Recommendation using textual opinions. In: IJCAI international joint conference on artificial intelligence, pp 2684–2690
14. Kim Y, Shim K (2014) TWILITE: a recommendation system for twitter using a probabilistic model based on latent Dirichlet allocation. *Inf Syst* 42:59–77
15. Zhang Y (2016) GroRec: a group-centric intelligent recommender system integrating social, mobile and big data technologies. *IEEE Trans Serv Comput* 9:786–795
16. Bhavitha B, Rodrigues AP, Chiplunkar NN (2017) In Comparative study of machine learning techniques in sentimental analysis. In: 2017 International conference on inventive communication and computational technologies (ICICCT), IEEE, pp 216–221
17. Salas-Zárate MdP, Medina-Moreira J, Lagos-Ortiz K, Luna-Aveiga H, Rodriguez-Garcia MA, Valencia-García R (2017) Sentiment analysis on tweets about diabetes: an aspect-level approach. In: Computational mathematical methods in medicine 2017
18. Huq MR, Ali A, Rahman A (2017) Sentiment analysis on Twitter data using KNN and SVM. *Int J Adv Comput Sci Appl* 8(6):19–25
19. Pinto D, McCallum A, Wei X, Croft WB (2003) In table extraction using conditional random fields. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, pp 235–242
20. Soni S, Sharaff A (2015) In sentiment analysis of customer reviews based on hidden Markov model. In: Proceedings of the 2015 international conference on advanced research in computer science engineering & technology (ICARCSET 2015), pp 1–5
21. Zhang X, Zheng X (2016) In comparison of text sentiment analysis based on machine learning. In: 2016 15th international symposium on parallel and distributed computing (ISPDC), IEEE, pp 230–233
22. Malik V, Kumar A (2018) Sentiment analysis of twitter data using naive Bayes algorithm. *Int J Recent Innov Trends Comput Commun* 6(4):120–125
23. Mehra N, Khandelwal S, Patel P (2002) Sentiment identification using maximum entropy analysis of movie reviews. Stanford University
24. Wu H, Li J, Xie J. Maximum entropy-based sentiment analysis of online product reviews in Chinese. In: Automotive, mechanical and electrical engineering
25. Firmino Alves AL, Baptista CdS, Firmino AA, Oliveira MGd, Paiva ACd (2014) A comparison of SVM versus Naïve Bayes techniques for sentiment analysis in tweets: a case study with the 2013 FIFA confederations cup. In: Proceedings of the 20th Brazilian symposium on multimedia and the web, ACM, pp 123–130
26. Pandey AC, Rajpoot DS, Saraswat M (2017) Twitter sentiment analysis using hybrid cuckoo search method. *Inf Process Manage* 53(4):764–779
27. Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: a survey. *Ain Shams Eng J* 5(4):1093–1113
28. Lu J, Wu D, Mao M, Wang W, Zhang G (2015) Recommender system application developments: a survey. *Decis Support Syst* 74:12–32
29. Betru BT, Onana CA, Batchakui B (2017) A survey of state-of-the-art: deep learning methods on recommender system. *Int J Comput Appl* 162(10)
30. Kardan AA, Ebrahimi M (2013) A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. *Inf Sci* 219:93–110
31. Nakayama M, Wan Y (2019) The cultural impact on social commerce: a sentiment analysis on Yelp ethnic restaurant reviews. *Inf Manag* 56:271–279
32. Jurafsky D, Chahuneau V, Routledge BR, Smith NA (2014) Narrative framing of consumer sentiment in online restaurant reviews. *First Monday*
33. Jia SS (2020) Motivation and satisfaction of Chinese and US tourists in restaurants: a cross-cultural text mining of online reviews. *Tour Manag* 78:104071
34. Meek S, Wilk V, Lambert C (2021) A big data exploration of the informational and normative influences on the helpfulness of online restaurant reviews. *J Bus Res* 125:354–367
35. Tian G, Lu L, McIntosh C (2021) What factors affect consumers' dining sentiments and their ratings: evidence from restaurant online review data. *Food Qual Prefer* 88:104060

36. Li H, Qi R, Liu H, Meng F, Zhang Z (2021) Can time soften your opinion? The influence of consumer experience valence and review device type on restaurant evaluation. *Int J Hosp Manag* 92:102729
37. Saini GL, Panwar D, Singh V (2021) Software reliability prediction of open source software using soft computing technique. *Recent Adv Comput Sci Commun (Formerly Recent Patents Comput Sci)* 14(2):612–621
38. Saini GL, Panwar D, Kumar S, Singh V, Poonia RC (2021) Predicting of open source software component reusability level using object-oriented metrics by Taguchi approach. *Int J Software Eng Knowl Eng* 31(02):147–166
39. Pankwar D, Saini GL, Agarwal P, Singh P (2022) Firefly optimization technique for software quality prediction. In: *Soft computing: theories and applications: proceedings of SoCTA 2021*. Springer Nature Singapore, Singapore, pp 263–273
40. Panwar D, Saini GL, Agarwal P (2022) Human eye vision algorithm (HEVA): a novel approach for the optimization of combinatorial problems. In: *Artificial Intelligence in Healthcare*, pp 61–71.

# Author Index

## A

- Aditya Ranade, 457  
Ahmed S. Alzuhairi, 245  
Aleksandra Ferens, 259  
Alfred Kirubaraj, A., 275  
Alomgir Hossain, 499  
Anand Nayyar, 363  
Anjusha Asok, 389  
Andrew, J., 201  
Ankur Chaurasia, 375  
Ankush Kumar, 215  
Anudeep Arora, 229  
Anu, K.P., 109  
Anuneshwar, 311  
Arunkumar Bongale, 95

## B

- Beata Dratwińska-Kania, 259  
Bibal Benifa, J.V., 109

## C

- Chandrasekhar Rao, D., 443

## D

- Daniel, D., 311  
Deepa, J., 405  
Dhilsath Fathima, M., 405

## G

- Garima Bisht, 81  
Gokulapriya, R., 311  
Gouri Goyal, 325

Gunjan Chugh, 325

## H

- Haider Al-Kanan, 245  
Hariharan, R., 405  
Harish, R., 339  
Harleen Kaur, 153  
Haya A. Alhakbani, 15

## J

- Jayanthi, K., 405  
Jeffin Joseph, 275  
Jenefa, J., 311  
Jino Ramson, S. R., 275  
Joseph Varghese Kureethara, 389  
Julian Benadit, P., 295

## L

- Latit Garg, 125, 139  
Leonardo Freitas, 125

## M

- Majid Bashir Malik, 473  
Makri, Eleni G., 417  
Mallikharjuna Rao, K., 153  
Manbha Kharsiyemlieh, 405  
Manjunath Ramanna Lamani, 295  
Mapreet Kaur Khurana, 41  
Md Asif Jamal, 1  
Md Iftekhar Ahmad, 287  
Mohd Ali, 473  
Momotaz Begum, 499

Monika Roopak, 229

## N

Namrata Gawande, 65  
 Nareshkumar, R., 53  
 Nasim Akhtar, 499  
 Navneet Bhargava, 41  
 Nikhil Jain, 375  
 Nimala, K., 53  
 Nitin Pise, 457

## P

Pal, A. K., 81  
 Payal Gupta, 375  
 Piyush Aggarwal, 325  
 Pradeep Kumar, 1  
 Priyanka Narad, 375

## R

Raghavendra Rao, G. N., 339  
 Ramesh, B. T., 95  
 Ranjeeta Kaur, 229  
 Rohit Mittal, 517

## S

Sadiya Bashir, 473  
 Sahoo, Himanshu Bhusan, 443  
 Saini, Ashok Kumar, 229  
 Saini, G. L., 517  
 Saini, Madan Lal, 215  
 Sambandam, Rakoth Kandan, 311  
 Sambhu Prasad, S., 353  
 Samruddhi Multaikar, 65  
 Sanjam Kaur Bedi, 153  
 Santwana Gudadhe, 191  
 Sarita Kumari, 431  
 Sarmah, Jibok, 215  
 Saurabh Kumar, 29  
 Saxena, Rahul, 201

Sayyad, Javed, 95  
 Sengupta, Abhishek, 375  
 Senith, S., 275  
 Shahid Mohammad Ganie, 473  
 Sharma, Jankiballabh, 487  
 Sharma, Shweta, 41  
 Sheena Kurian, K., 173  
 Sheena Mathew, 173  
 Shekhawat, Sayar Singh, 229  
 Shome, Sachi, 405  
 Shrivastava, Vishal, 517  
 Singh, Pardeep, 125, 139  
 Sirisha, D., 353  
 Soni, Nainsi, 29  
 Srushti Patil, 65  
 Subodh Kumar, 353  
 Sumit Kumar, 517  
 Suvvari, Somaraju, 287

## T

Tanvi Mehta, 65  
 Thahab M. AlBuhairi, 15  
 Thakare, Anuradha, 191  
 Tomer, Neha, 229  
 Tripti Lamba, 325  
 Truong, Van-Truong, 363

## U

Upadhyaya, Amrita, 431  
 Utkarsh Rastogi, 95

## V

Vaibhav Kumar, 325  
 Vassallo, Michael, 139  
 Vats, Prashant, 229  
 Vetriveeran, Divya, 311  
 Vibhakar Pathak, 517  
 Vidhan Chasta, 215  
 Vijay Prakash, 125, 139  
 Vyas, Arpita, 487