Sridaran Rajagopal · Kalpesh Popat ·
Divyakant Meva · Sunil Bajeja ·
Pankaj Mudholkar (Eds.)

Communications in Computer and Information Science 2426

Artificial Intelligence Based Smart and Secured Applications

Third International Conference, ASCIS 2024
Rajkot, India, October 16–18, 2024
Revised Selected Papers, Part III

Part 3

Springer

Communications in Computer and Information Science 2426

Series Editors

Gang Li, *School of Information Technology, Deakin University, Burwood, VIC, Australia*
Joaquim Filipe, *Polytechnic Institute of Setúbal, Setúbal, Portugal*
Zhiwei Xu, *Chinese Academy of Sciences, Beijing, China*

Rationale
The CCIS series is devoted to the publication of proceedings of computer science conferences. Its aim is to efficiently disseminate original research results in informatics in printed and electronic form. While the focus is on publication of peer-reviewed full papers presenting mature work, inclusion of reviewed short papers reporting on work in progress is welcome, too. Besides globally relevant meetings with internationally representative program committees guaranteeing a strict peer-reviewing and paper selection process, conferences run by societies or of high regional or national relevance are also considered for publication.

Topics
The topical scope of CCIS spans the entire spectrum of informatics ranging from foundational topics in the theory of computing to information and communications science and technology and a broad variety of interdisciplinary application fields.

Information for Volume Editors and Authors
Publication in CCIS is free of charge. No royalties are paid, however, we offer registered conference participants temporary free access to the online version of the conference proceedings on SpringerLink (http://link.springer.com) by means of an http referrer from the conference website and/or a number of complimentary printed copies, as specified in the official acceptance email of the event.

CCIS proceedings can be published in time for distribution at conferences or as post-proceedings, and delivered in the form of printed books and/or electronically as USBs and/or e-content licenses for accessing proceedings at SpringerLink. Furthermore, CCIS proceedings are included in the CCIS electronic book series hosted in the SpringerLink digital library at http://link.springer.com/bookseries/7899. Conferences publishing in CCIS are allowed to use Online Conference Service (OCS) for managing the whole proceedings lifecycle (from submission and reviewing to preparing for publication) free of charge.

Publication process
The language of publication is exclusively English. Authors publishing in CCIS have to sign the Springer CCIS copyright transfer form, however, they are free to use their material published in CCIS for substantially changed, more elaborate subsequent publications elsewhere. For the preparation of the camera-ready papers/files, authors have to strictly adhere to the Springer CCIS Authors' Instructions and are strongly encouraged to use the CCIS LaTeX style files or templates.

Abstracting/Indexing
CCIS is abstracted/indexed in DBLP, Google Scholar, EI-Compendex, Mathematical Reviews, SCImago, Scopus. CCIS volumes are also submitted for the inclusion in ISI Proceedings.

How to start
To start the evaluation of your proposal for inclusion in the CCIS series, please send an e-mail to ccis@springer.com.

Sridaran Rajagopal · Kalpesh Popat ·
Divyakant Meva · Sunil Bajeja ·
Pankaj Mudholkar
Editors

Artificial Intelligence Based Smart and Secured Applications

Third International Conference, ASCIS 2024
Rajkot, India, October 16–18, 2024
Revised Selected Papers, Part III

Springer

Editors
Sridaran Rajagopal
Marwadi University
Rajkot, Gujarat, India

Kalpesh Popat
Marwadi University
Rajkot, Gujarat, India

Divyakant Meva
Marwadi University
Rajkot, Gujarat, India

Sunil Bajeja
Marwadi University
Rajkot, Gujarat, India

Pankaj Mudholkar
Marwadi University
Rajkot, Gujarat, India

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-031-86295-3 ISBN 978-3-031-86296-0 (eBook)
https://doi.org/10.1007/978-3-031-86296-0

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2025

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 Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

Preface

"A Little Progress Each Day, Adds up to a Big Result"

Our 3rd International Conference on Advancements in Smart Computing & Information Security (ASCIS 2024) invited original research papers under five tracks, namely, **AI & ML, Smart Computing, Cyber Security, Networking and Cloud Computing and Computer Applications for Sustainability.** We received 667 submissions and 181 were recommended for publication in Springer's CCIS series. The papers will appear as 6 volumes (Volume Numbers ranging 2424–2429). We are very much thankful to the editorial board of **Springer CCIS** for their continuous support.

The submitted papers were involved in an open review process, involving three reviewers for each paper. We are thankful to the Technical Program Committee (TPC) members who were very supportive in the review process.

Several sponsors who contributed generously towards the successful running of ASCIS 2024, including **Science and Engineering Research Board (SERB), K7 International and D- Link Academy**. These collaborations enriched the conference and extended its outreach.

ASCIS 2024 was conducted during 16–18 October, 2024, and featured insightful keynote addresses from national and international experts in their respective fields. The line-up included **Prabir Kumar Biswas** from IIT Kharagpur, Cybersecurity expert **Ram Kumar G, Arnav Bhavsar** from IIT Mandi, **Murali Raman** from Asia Pacific University, and **Aravind Benegal** from NLB Services.

The keynotes addressed several recent technological advancements whereas the hands-on workshops provided practical exposure to the students and participants. The PhD forum featuring the research works of research scholars and recent PhD graduates added value to the conference and acted as a platform for knowledge sharing.

These volumes titled **"Artificial Intelligence based Smart & Secured Applications"** are the consolidation of all such solutions employing technologies of AI, ML, Smart Computing and, Cybersecurity in every significant field seeking research solutions.

October 2024 Sridaran Rajagopal

Organization

General Chair

R. Sridaran	Marwadi University, India

Program Committee Chairs

R. Sridaran	Marwadi University, India
Kalpesh Popat	Marwadi University, India
Divyakant Meva	Marwadi University, India
Sunil Bajeja	Marwadi University, India
Pankaj Mudholkar	Marwadi University, India

Steering Committee

Ketanbhai Marwadi	Marwadi University, India
Jitubhai Chandarana	Marwadi University, India
P. Bhanu Prasad	Future Technology
R. B. Jadeja	Marwadi University, India
Sanjeet Singh	Marwadi University, India
Prabir Kumar Biswas	IIT Kharagpur, India
Anil Kumar Sao	IIT Bhilai, India
Arnav Bhavsar	IIT Mandi, India
Naresh Jadeja	Marwadi University, India
Shamala Subramanyam	Universiti Putra Malaysia, Malaysia
Padmavathi	Avinashilingam Institute for Home Science and Higher Education for Women, India
T. Devi	Bharathiar University, India
K. Iyakutti	SRM Institute of Science and Technology, India
Ram Kumar G.	Global Automotive Company, India

Technical Program Committee Chairs

AI and ML Track

Hitesh Chhinkaniwala	Adani University, India
Ahlad Kumar	National Forensic Sciences University, India
Mahendra Kumar Gourisaria	Kalinga Institute of Industrial Technology, India

Cyber Security Track

Ruchira Naskar	Indian Institute of Technology, Kharagpur, India
Shravani Shahapure	Deloitte Touche Tohmatsu, India
Ritika Vivek Ladha	Adani University, India

Smart Computing Track

Manish Khare	Dhirubhai Ambani Institute of Information and Communication Technology, India
Dinesh Kumar	Bennett University, India
Rebakah Geddam	Nirma University, India

Networking and Cloud Computing Track

Rachna Jain	Bhagwan Parshuram Institute of Technology, India
Santosh Kumar Majhi	Guru Ghasidas Vishwavidyalaya (Central University), India
Nishant Doshi	Pandit Deendayal Energy University, Gandhinagar, India

Computer Applications for Sustainability Track

Manas Ranjan Pradhan	Skyline University College, UAE
J. Ramkum	Krishna Arts and Science College, India

Additional Reviewers

Arun Raj Lakshminarayanan
Dhanamma S. Jagli
Tushar H. Jaware
Kamal Sutaria
Mahalakshmi G.
Nebojsa Bacanin
Rahul Ashok Patil
Satvik Khara
Vijay D. Katkar
Vipin Sharma
Amarinder Kaur
Anita Gautam Khandizod
Bhoopendra Singh
Delecta Jenifer Rajendren
Harishchander Anandaram
Karuna Nidhi Pandagre
Amrita Kumari
Dipti H. Domadiya
Prashant Sen
Rajasekaran Selvaraju
Saran Raj Sowrirajan
Harshal Anil Salunkhe
Sumathy S.
Hemant Ingale
Jamberi K.
Jayashree Nair
Jyoti Kharade
Keerti Jain
Kumaresh N.
Mythili Shanmugam
Nikita Bhatt
Parvathaneni Naga Srinivasu
Praveen P. V. Sundar
Ravirajsinh S. Vaghela
Richa Adlakha
Ritesh Patel
Sharanyaa S.
Srinivasulu Raju S.
Swetta Kukreja
Vinothina V.
Ajay Kushwaha
Anbumani K.
Ashwin B. Makwana
Chandresh K. Kumbharana
Divyakant T. Meva
Raji C. G.
Anita M. Indu
Kailash Patidar
Kannadhasan Suriyan
Hardik K. Molia
Jayashri Pattar
Jonnadula Narasimharao
Kaviyarasi R.
Ramakrishna Garine (Ram)
Nageswari D.
Prema C.
Rajender Kumar
Ram Ratan
S. Amuth Sachin H. Gajjar
Sreenaga Chidambara Vinjamuri Dattu
Vipul Shah
A. Yovan Felix
Abhay Singh Bhadauria
Aditi Sharma
Ajay Kumar Kushwaha
Ajita Deshmukh
Akash Saxena
Amita Sharma
Ankit Subhash Didwania
Anubhav Kumar Prasad
Arun Adiththan
Asha V.
Ashokkumar Baldevbhai Prajapati
Avinash Sharma
Bhagvati Parekh
Brijesh R. Jajal
Dafni J. Rose
Dhivya K.
Dipti Chauhan
Divya R.
Divyanshu Chandra
Ashwin R. Dobariya
Prabhat Sharma
Rohit Kanauzia
Abhinav Tomar
Akshara Dave

Ashish Saini
Avnip Deora
Dimple M. Thakar
Gaurav Kumar Ameta
Jayant R. Nandwalkar
Kapil Joshi
Lokesh P. Gagnani
Nisha Khurana
Samir B. Patel
Sri Hari Nallamala
Sunil L. Bajeja
T. Sathish Kumar T.
Parwinder Kaur
Rajesh Bansode
Saraswathi S.
T. Buvaneswari Muruganantham
Vijayalakshmi P. S.
Galiveeti Poornima
Hari Kumar Palani
Hemraj S. Lamkuche
Jafar Ali Ibrahim Syed Masood
Jay A. Dave
Jebakumar Immanuel D.
Juliet Rozario
Kajal S. Patel
Karthikeyan Eswaramurthy
Kavitha Ganesh
Krishnakumar A.
Lata Suresh
Latchoumy P.
M. R. Ramesh
Madhu Shukla
Manisha Rawat
Maruthamuthu R.
Maulika Patel
Megha P. Mudholkar
Mohamed Iqbal M.
Mohit Tiwari
Muralidharan R.
Narayan Joshi
Neeru Sharma
Neha Sharma
Nidhi M. Patel
Nilu Singh
Pankaj K. Mudholkar
Parth Gautam
Pranav Verma
Praveen Kumar
Priyanka Sharma
Purnendu Bikash Acharjee
Radha B.
Rajesh Yadav
Rakesh Kumar Yadav
Renjith V. Ravi
Richa Ashok Modiyani
Rohit Goyal
Rupali Atul Mahajan
S. S. Priscila
Saifullah Khalid
Sandeep Mathur
Saswati Mahapatra
Senthil Kumar R.
Shamala K. Subramaniam
Shanti Verma
Shweta Sharma
Srinivasa Rao Pallapu
Subramanian Balambigai
Sudipta Hazra
Sunil Gautam
Suresh B. Kumar
Tanmay Kasbe
Thirumurugan Shanmugam
Uma N. Dulhare
Umesh S. Pinjarkar
Vaibhav C. Gandhi
Vandani Verma
Vipin Sharma
Vishvjit K. Thakar
Yogesh Ghodasara
Chandran C. P.
Raghu N.
Jaimin N. Undavia
Kavipriya P. P.
Mohamed Mosbah
Padma Selvaraj
Remegius Praveen Sahayaraj
Shadab Siddiqui
Vineet Kumar Singh
Abhishek Sharma
Anilkumar Suthar

Bharanidharan G.
Chandra J.
Disha J. Shah
John T. Abraham
P. Rizwan Ahmed
B. Surendiran
Dushyantsinh B. Rathod
Priya K.
Rushikumar R. Raval
Ajitha V.
Kajal Patel
Gourav Gupta
J. Grace Hannah
Jasminder Kaur Sandhu
Jinal H. Tailor
Karthik B.
Kruti K. Sutaria
K. P. Malarkodi
Naresh Kumar
Nirav Bhatt
Pradip Mathuradas Jawandhiya
Rajeswari S.
Revathy G.
Ripal D. Ranpara
Senthilkumar Meyyappan
Shikha Maheshwari
Sweeti Sah
Vijay Karnatak
A. Maheswary
Anamika Rana
Anjali Diwan
Biswaranjan Mishra
Chintan Patel
Himanshu K. Maniar
T. S. Murugesh
Darshankumar C. Dalwadi
Anitha K.
Priya Chandran
Jalpesh Vasa
Jeevitha R. R.
Jose M. Molina
Lilly Florence M.
Mohan Subramani
Naresh Kshetri
Qixia Zhang
Rajeswari M.
Roshan Anant Gangurde
Safvan Vahora
Vijaya Kittu Manda
Yugendra Devidas Chincholkar
A. P. Nirmala
Abhilasha Vyas
Ahmed Banimustafa
Ajay M. Patel
Ajitha I.
Akoramurthy Balasubramaniam
Angeline R.
Ankur N. Shah
Anup Palsokar
Aruna Animish Pavate
Ashish Kumar
Ashwin Raiyani
Balraj Verma
Boopathi Raja G.
Chandra Mohan
Darshita S. Pathak
Dipak Ramoliya
Divya Didwania
Anant G. Kulkarni
Chintan B. Thacker
Deepak Kumar Verma
Rashmi Soni
Saraswathi S.
Abhishek Sharma Padmanabhan
Ankit J. Faldu
Asmita Manna
Daxa Vekariya
Disha H. Parekh
Jay Kumar Jain
Jaypalsinh A. Gohil
Krupa S. Mehta
Nafees Akhter Farooqui
Premkumar Borugadda
Rupesh Kumar Jindal
Shadab Siddiqui
Sudhanshu Maurya
Sunil Gupta
Vatsal H. Shah
Rajan Patel
Ramesh T. Prajapati

Sreejith Vignesh B. P.
Veena Soni
G. Charles Babu
Gaurav Agrawal
Harish Padmanaban
Himanshu Rai
Jane K. Nithya
Jaydeep R. Ramani
Juhi Singh
Jyotirmoy Pathak
Kamal Saluja
Karthikeyan R.
Kedir Lemma Arega
Kumuthini C.
Lataben J. Gadhavi
Lipsa Das
M. Vinoth Kumar
Mahalakshmi G.
Manoj Kr. Mishra
Mastan Vali Shaik
Meet Patel
Minal S. Shukla
Mohammed Wajid Khan
Monther Ali Tarawneh
Nagaraju Kilari
Neha Parashar
Neha Ripal Soni
Nilesh Vijay Sabnis
Pagalla Bhavani Shankar
Paresh V. Virparia
Pragadeswaran S.
Prashant P. Pittalia
Preethi S. R.
Priyanka Suyal
Pushparaj Pal
Rajib Biswas
Ramkumar Jaganathan
Rinkoo Bhatia
Ruchika Pharswan
Rutvi R. Shah
Sadhana Singh
Samriti Mahajan
Sanjeevkumar D. Angadi
Savitha Jaganathan
Shafi Pathan
Shantha Mary Joshitta
Shilpa Mehta
Sivanesan Rajangam
Subhadip Sarkar
Sudha R.
Suhasini Vijaykumar
Sunil Saxena
Surya Kameswari Uduga
T. D. Sudhakar
Tarannum Bloch
Tripti Tiwari
Umang Thakkar
Vaibhav A. Gandhi
Vallidevi Krishnamurthy
Vikas Tripathi
Vipul A. Shah
Yogendra Kumar
Chetan Dudhagara

Our Sponsors

Sr. No.	Name of the Sponsor	Logo
1	SERB India	
2	D-Link	
3	Digital India	
4	K7 International	
5	Samatrix.io	
6	Marwadi Technologies	
7	Stelcore	

Abstracts of Keynotes

Transforming Industry 4.0 with Generative AI and Cloud Computing

Gundarapu Pavan

Safran DIFIT, France

Industry 4.0, characterized by the convergence of automation, IoT, and advanced analytics, is reshaping industries globally. As organizations strive for greater efficiency, innovation, and adaptability, the synergy between cloud computing and generative AI is emerging as a transformative force. This keynote will explore the vital role of cloud computing as the backbone of Industry 4.0, providing scalable infrastructure and real-time data accessibility. It will delve into what generative AI is, how it acts as a catalyst for innovation, and how it accelerates the development of intelligent systems, from predictive maintenance to automated design and decision-making. By examining the synergies between these two technologies, the keynote will illustrate their potential to revolutionize industrial operations and unlock new business value.

Attendees will gain insights into the tangible benefits and ROI of integrating cloud and AI solutions. The keynote will also address the challenges organizations face—ranging from data security to infrastructure optimization—and offer strategic considerations for successful implementation. Looking ahead, we'll explore future trends in Industry 4.0, where cloud-enabled generative AI could redefine the industrial landscape, driving sustainable growth and continuous innovation.

The session will conclude with a call to action, encouraging students and industry leaders to embrace these technologies to remain competitive in a rapidly evolving digital economy.

Brain Decoding with Deep Learning

Arnav Bhavsar

IIT, Mandi, India

The talk will focus on visual brain decoding via EEG using deep learning techniques. Brain decoding, also known as Perceptual Brain Decoding (PBD), involves using brain responses to different stimuli to identify the original stimulus or its characteristics.

Specifically, we target an external visual stimulus, which involves object images on a computer screen. The goal is to reconstruct the class of the image from the EEG data captured with observing the image. The decoding involves either classifying reconstructing the type of image shown during the visual task.

For reconstructing class-specific images, we will discuss the utilization of generative networks such as GANs, using embeddings from a pre-trained EEG classifier network as input. Additionally, we will also cover an end-to-end generative network (NeuroGAN) for image synthesis, where the generator produces class-specific embeddings and images from EEG signals.

Advanced Insights into Deep Fake Recognition

Deepak Kumar Jain

Dalian University of Technology, China

The rapid advancement of artificial intelligence and machine learning has led to the proliferation of deep fake technology, enabling the creation of highly realistic, AI-generated synthetic media. While deep fakes offer exciting possibilities in fields like entertainment and virtual reality, they also pose significant threats to public trust, cyber security, and political stability. This presentation, *Advanced Insights into Deep Fake Recognition*, explores the complex techniques used to produce deep fakes and highlights the growing challenges in detecting them. We delve into the mechanics of deep fake creation, particularly the role of Generative Adversarial Networks (GANs), and discuss the real-world implications of this technology's misuse, from identity theft to political disinformation.

The presentation also examines cutting-edge detection methods, including AI-driven facial and audio analysis, and emerging technologies like block chain and quantum computing, which are being employed to verify the authenticity of digital content. Furthermore, we explore current applications of deep fake detection tools across social media platforms, law enforcement, and the media, while considering the ethical and legal issues surrounding the use and regulation of deep fake technology. By providing a comprehensive overview of the challenges and future trends in deep fake recognition, this presentation aims to equip audiences with the knowledge to navigate this evolving digital landscape.

Security Models and Strategies of Cloud Computing in Business and Public Administration

Zdzislaw Polkowski

WSG University, Poland

Globalization and the development of information and communication technologies (ICT) have caused significant changes in business and public administration in the European Union. Currently, the digital economy is driven by modern ICT systems that offer employers and employees new tools for effective operation. Cloud Computing systems deserve special attention, as they have an impact on business, public administration and the everyday life of citizens. Cloud Computing creates many benefits not only for users but also has a positive impact on the environment, which is in line with the & quot; European Green Deal & quot; strategy implemented in Europe. However, the security of Cloud Computing and many previously unseen threats are always in the focus of both cloud users and providers of this modern solution. Threats, such as zero-day exploits, advanced persistent threats (APTs), and insider attacks are some popular threats that strained the cloud services in business and public administration. These threats create a large barrier to widespread use and in many cases delay the further development of these systems. The regulations constantly introduced and updated in Europe take into account the development of secure Cloud Computing, sensibly regulating the ways of processing not only personal data, but also data processing in general. It is worth mentioning that research and analysis have been carried out in real companies and public administration institutions to build robust and secure cloud platforms that can tackle modern-day cyber threats. The analysis highlights that the best outcomes in cloud computing implementations are driven by the collaboration of interdisciplinary teams, specifically those composed of IT specialists, legal experts, and managers. This synergy ensures a holistic approach, balancing technical efficiency with legal compliance and strategic oversight. Special attention should be given to security strategies, such as robust data encryption, identity and access management, regular auditing, and compliance management to mitigate the negative consequences of security breaches in cloud environments. These advancements will be particularly beneficial for a wide audience, including IT professionals, researchers, students, and business leaders, all of whom play a role in implementing and managing cloud solutions. Furthermore, future research could explore how these interdisciplinary teams tackle emerging challenges, such as evolving regulatory landscapes, data sovereignty, and the integration of advanced technologies like AI and machine learning in cloud infrastructure. Such investigations would contribute to the ongoing development of best practices and position cloud computing as a more secure and sustainable solution in a rapidly changing digital ecosystem.

Keywords: Cloud Computing · IT security · Business · Public administration · European Green Deal

Role of Computing and Technology in Sustainable Development Goals

Murali Raman

Asia Pacific University of Malaysia

We are living in a highly Volatile, Uncertain, Complex and Ambiguous (VUCA) world, which threaten the long-term sustainability of global business, economy and natural living. The Sustainable Development Goals (SDGs) outlined by the United Nations serve as a global roadmap for achieving a more equitable and sustainable future. As the world grapples with pressing challenges such as climate change, poverty, and inequality, computing and technology have emerged as powerful tools for addressing these complex issues. This keynote explores the pivotal role of computing and technology in advancing the SDGs, highlighting specific applications and their potential impact.

Computing and technology can contribute to the SDGs in various ways. For instance, in the realm of climate change, advanced analytics can be used to optimize energy consumption and identify sustainable energy sources. Artificial intelligence can enhance the efficiency of renewable energy systems and predict natural disasters, enabling early warning systems and disaster preparedness. Additionally, computing-powered solutions can facilitate sustainable agriculture by optimizing crop yields, reducing water consumption, and minimizing the use of harmful chemicals.

In the context of poverty and inequality, computing and technology can empower marginalized communities through digital inclusion and education. Affordable and accessible internet connectivity can bridge the digital divide, providing opportunities for education, healthcare, and economic development. Furthermore, online platforms can facilitate access to financial services, enabling individuals and small businesses to escape poverty.

Computing and technology can play a crucial role in promoting good health and well-being. Telemedicine and remote healthcare services can improve access to quality healthcare, especially in rural and underserved areas. Additionally, data analytics can be used to identify disease outbreaks and track their spread, enabling effective public health interventions.

In summary, computing and technology offer immense potential for addressing the SDGs. By harnessing the power of these tools, we can create a more sustainable, equitable, and prosperous world for all. This keynote uses case examples from different parts of the world, showcasing how SDGs can be addressed with proper utilisation of computing and technology. However, realizing this potential requires collaboration among governments, businesses, and academia to develop innovative solutions and ensure equitable access to technology.

Cybersecurity in the Age of 4.0 IR

Srinath Doss

Botho University, Botswana

In the era of Industry 4.0, where cyber-physical systems, the Internet of Things (IoT), and artificial intelligence drive unprecedented connectivity and automation, cyber security has never been more critical. As industries leverage smart technologies to enhance productivity, efficiency, and innovation, the digital landscape expands, bringing both opportunities and vulnerabilities.

The convergence of operational technology with information technology in smart factories, autonomous vehicles, and interconnected supply chains creates a complex and dynamic environment. A single security breach can disrupt production, compromise sensitive data, and even endanger lives. Cyber attacks are evolving in sophistication, targeting the very backbone of our digital infrastructure.

Cybersecurity in the age of Industry 4.0 demands a proactive and multi-layered approach. It requires the integration of advanced threat detection systems, robust encryption protocols, and comprehensive risk management strategies. Organizations must foster a culture of cybersecurity awareness, ensuring that employees at all levels understand and adhere to security best practices.

Collaboration across sectors is essential. Governments, private enterprises, and academic institutions must work together to develop innovative solutions, share threat intelligence, and establish global cybersecurity standards. Continuous investment in research and development is crucial to staying ahead of cybercriminals and safeguarding our digital future. In this digital revolution, cybersecurity is not just a technical necessity but a fundamental pillar of trust and resilience. By embracing a holistic and forward-thinking approach, we can protect our assets, secure our data, and pave the way for a safer, more interconnected world.

Keywords: Industry 4.0 · Cyber security · Internet of things · Cyber Attack · Risk Management

Trust in the Realm of Generative AI – Opportunities and Threats

Arvind Benegal

Chief Risk Officer & Sr. Advisory Consultant, USA

Trust is of paramount importance in Information Technology systems and data confidentiality, integrity and availability. AI and Generative AI have reshaped the technology landscape. A balanced approach to innovation is needed that incorporates both Trust and Risk Management. The threats represent the clear and present risks and challenges posed by emerging technologies, while the opportunities speak to the benefits brought by these innovations. This highlights the dual nature of technological advancement. Wherever there is an opportunity, there is also a significant threat.

The opportunities include the efficiency gains of automation of repetitive tasks and enhanced decision-making. Personalization addresses tailored experiences for users in healthcare, finance, and customer service. Innovation allows for creation of new products, services, and business models. Democratization of technologies provides access, challenging elite layers, and Just in Time Knowledge suggests just enough to solve problems.

The risks include Bias and Fairness since AI systems may perpetuate or amplify existing biases. Security vulnerabilities pose increased attack surfaces and potential misuse of AI capabilities. Data Privacy presents risks of unauthorized data usage and breaches, and Prompt Injections increase exposure to sophisticated AI attacks.

The Audit Focus Areas should consider Data Integrity. It must be ensured that data used by AI systems is accurate, complete, and secure, and ethical AI establishes frameworks to prevent bias and ensure fairness. Accountability and transparency are achieved by creating mechanisms for auditing AI decisions and actions. Institutional guidance provides adherence to IT governance frameworks such as COBIT to manage AI risks. Robust control measures are implemented to enhance trust. For instance, ISO/IEC 42001 is an international standard that provides a governance framework for implementing and continually improving artificial intelligence management systems.

Trust is built by by incorporating strategies for IT Auditors. Continuous Monitoring regularly reviews AI systems and processes for compliance and performance. Collaboration with stakeholders by engaging with developers, data scientists, and business leaders aligns AI initiatives with ethical standards. Education and Awareness train employees on the risks and best practices related to AI. And finally, a Risk-Based Approach is adopted by prioritizing audits and controls based on the risk levels associated with different technologies.

Machine Learning and Data Analytics at Scale - Rise of Unified AI and Data Platforms

Kumar Gangwani

BDB.ai, India

The last 10+ years have seen a breakneck pace of innovation in areas of Data Analytics, Machine Learning and AI, something that remained unseen in many decades prior. The availability of inexpensive storage, affordable compute, innovations in parallel computing frameworks, global availability of open-source tools and libraries, automation of data engineering, algorithmic innovations and availability of specialized hardware like GPUs and TPUs have ushered in the modern Data and AI era.

The open-source ecosystem has democratized access to cutting-edge tools and technologies to the larger technology community. For instance, frameworks like TensorFlow and PyTorch allow developers and researchers to build sophisticated models quickly and efficiently.

Yet, most organizations, despite good intentions, sizeable investments in technology and large technology teams, struggle to scale after successfully operationalizing a few initial analytics and ML use cases. In fact, the rate of failures of corporate data science and machine learning teams is so high that the notion of the data and AI-driven enterprise often seems a myth.

Despite the availability of a plethora of tools for data engineering and machine learning, most organizations fail to build and scale with these effectively. Part of this has to do with the overpromise around capability of enterprise tools sold to them, which sometimes are good at a particular solution like ingestion or discovery only; this results in organizations purchasing a number of tools that don't integrate well with each other or scale for increasing data volumes and diverse data demands. But that's not the only reason. Fragmented, inconsistent and incomplete data is often a factor responsible for failure of Machine Learning projects. Integrating ML workflows into production environments, handling versioning, scaling models and automating monitoring – a discipline being referred to as ML Ops – remain challenging for many businesses; and add to that rising complexity of algorithms itself. Most technical resources have specialization in perhaps one or two areas, but a very limited view of the entire data chain, and struggle to solve platform scale issues.

The whole industry today longs for an integrated approach that allows operation of analytics and ML at scale. The past few years have seen the rise of a few unified platforms that integrate the entire data infrastructure and plethora of data processes – from ingestion (batch, micro batch, real-time), data cleansing, enrichment and preparation, storage into data lake, with right security and governance, notebook/tools to experiment with and deploy Data Science and ML models, enable ML automation, use of AI services and self-service dashboards and visualization – all under one roof. These unified platforms

greatly simplify and accelerate the time to commercialize analytics use cases, often 4–5 times faster than commonly used tools, and the end-to-end automation of all data processes greatly reduces resource effort by a large margin, providing for sizeable cost savings. This integrated capability offered by unified data and AI platforms is the future of enterprise analytics and Machine Learning.

Implementing Cyber Security Best Practices to Protect Your Crown Jewels

Ram Kumar G.

Global Automotive Company, India

In a hyper-connected digital world, cyber threats are ever growing and evolving, putting confidential data supporting IT infrastructure at risk. Cyber crimes are common today and no organization is immune to cyber attacks. The cyber crime industry has grown into a multi-billion dollar illegal business and the profile of the attacker has changed over the years – from script kiddies to well-organized hacker groups supported by nation states.

In this prevailing scenario, it is important for organizations to be cognizant of the cyber threats and proactively put in place cyber security measures by way of security controls, processes backed up by skilled talent. A strong understanding of the multi-faceted domains of cyber security is essential to appreciate the range and depth of the field. This exposure leads to exploring the security best practices that need to be implemented in organizations as per their risk profile to safeguard their critical IT infrastructure and confidential business information including intellectual property in what has come to be characterized as "crown jewels". The security controls are defined by the results of a risk assessment exercise, situational awareness of ongoing cyber attack campaigns enabled by inputs from cyber threat intelligence, obligations to customers and regulatory requirements for cyber security and data privacy.

Keywords: Cyber security · Best practices · Crown jewels · Cyber attacks · Data protection · Critical assets · Cyber threats · Security risks

Smart Spatial Computing and Its Application

P. Bhanu Prasad

Future Technology, France

Spatial computing is an emerging technology that blends digital and physical worlds, allowing users to interact with smart devices in more seamless and immersive ways. It leverages a range of technologies, including augmented reality (AR), extended reality (XR), Artificial intelligence, Internet of things (IoT), haptic feedback, computer vision, sensor fusion, geospatial data analysis, and spatial mapping. Spatial computing integrates various technologies to interpret the physical world and merge it with digital information, effectively combining both realms. These experiences can range from fully immersive environments to the addition of digital elements in the real world, creating interactive experiences that enhance our understanding of spatial relationships. At its core, spatial computing empowers users to visualize, manipulate, and analyze complex spatial data in real time with ease.

Advancements in hardware and software enable developers to create rich, immersive, interactive environments that captivate users. This evolution not only improves the overall user experience but also opens up new possibilities. Despite the significant potential of spatial computing, challenges persist. Concerns regarding data privacy, security, and ethical considerations need to be addressed before these technologies can be widely deployed. Additionally, the lack of standardization and interoperability among different spatial computing platforms presents further obstacles to widespread adoption.

This immersive, natural way of interacting can enrich our lives in numerous ways, enhancing hobbies like entertainment, gaming, and sports, as well as educational experiences and professional activities. The applications of spatial computing are extensive and diverse, impacting vital sectors that shape daily life, including navigation, astronomy, virtual home design, learning through gaming, skill development, healthcare and fitness, virtual shopping, industrial settings, remote assistance, architecture and design, the stock market, and logistics and supply chain management.

Spatial computing signifies a paradigm shift in how we interact with various environments, unlocking a vast array of new applications. Here are some key applications of spatial computing that can help those interested in pursuing a career in this field get started.

Keywords: Spatial computing · Extended reality (XR) · Industrial Internet of Things (IIoT) · Sensor fusion, Digital Twins · Artificial intelligence, Haptic

Enhancing Recommender Systems with Extensions of Fuzzy Sets for Considering Stakeholders' Uncertainty in Software Requirements Prioritization

Vassilis C. Gerogiannis

University of Thessaly, Larissa, Greece

In large, complex and distributed software development projects, it is often not feasible to implement all of the requirements in a single software release because of budget, time and staff constraints. Software requirements prioritization is an important activity in software requirements engineering which aims to evaluate and prioritize the candidate software features to be implemented in the next release of a software system. Requirements prioritization is often applied iteratively, according to various criteria, by the multiple involved project stakeholders who may have different roles, needs and knowledge. In large-scale software development projects, it is often practically impossible to ensure that all stakeholders have complete knowledge on all candidate requirements. Thus, stakeholders may show some degree of uncertainty and hesitation, as it is difficult to precisely evaluate each requirement according to each prioritization criterion. This talk will present results of our ongoing research work on combining Recommender Systems and soft computing techniques based on extensions of Fuzzy Sets to address problems of requirements prioritization in the context of large and complex software development/maintenance projects. Recommender Systems can be a useful solution to information overload situations when stakeholders have to evaluate a large number of alternatives. Fuzzy sets extensions, such as Intuitionistic Fuzzy Sets, can deal with stakeholders' uncertainty and hesitation regarding the prioritization criteria importance and requirements ratings. The results from applying our methods are promising since they demonstrate that they can be effectively used to support multiple stakeholders to prioritize large requirements sets.

Keywords: Software Requirements Prioritization · Recommender Systems · Intuitionistic Fuzzy Sets

Generative Adversarial Network (GAN): An Efficient Tool for Data Generation and Data Completion

Prabir Kumar Biswas

IIT, Kharagpur, India

Generative Adversarial Network (GAN) is a relatively recent concept in deep generative learning wherein two neural networks are pitted against each other in a zero-sum non-cooperative game to match a non-stationary distribution to an intractable stationary distribution. In this lecture we shall talk about how the generative modelling capability of GAN can be exploited to design efficient frameworks for image/video inpainting. Semantic inpainting refers to realistically filling up large holes in an image or video frame. Our approach is to first train a generative model to map a latent noise distribution to a natural image manifold and, during inference time, search for the best-matching noise vector to reconstruct the signal. The primary drawback of this approach is its inference time iterative optimization and lack of photo-realism at higher resolution. In this talk both of the abovementioned shortcomings are addressed. This is made possible with a nearest neighbour search-based initialization (instead of random initialization) of the core iterative optimization involved in the framework. The concept is extended for videos by temporal reuse of solution vectors. Significant speedups of about 4.5-5x on images and 80x on videos is achieved. Simultaneously, the method achieves better spatial and temporal reconstruction qualities.

Next, we will pose the following question and try to get the answer - 'Do we at all need an iterative inference framework?' - to answer this, a data-driven parametric network is trained to directly predict a matching prior for a given masked image. This converts an iterative paradigm to a single feed-forward inference pipeline with around $800\times$ speedup. Finally, recent advancements in high-resolution GAN training are leveraged to scale an inpainting network for higher resolution.

Artificial Intelligence and Machine Learning: From Early Symbolic Systems to More Complex System

Salisu Mamman Abdulrahman

Kano University of Science and Technology, Nigeria

Artificial Intelligence (AI) and Machine Learning (ML) have evolved significantly over the years, moving from simple symbolic systems to more complex and sophisticated systems capable of learning and adapting. Advances in AI and ML systems have revolutionized various sectors, changing how people interact with technology. For four decades (mid-1950s until the mid-1990s), Symbolic AI was the dominant paradigm. These algorithms work by processing symbols that represent objects or concepts in the real world and their relationships. In Symbolic AI the main approach is to use logic-based programming where rules and axioms are used to perform inference and deduction. High expectations surrounding AI's potential to solve complex problems and mimic human intelligence were met with disappointing results, leading to what is known as the AI Winter. Despite the challenges faced by researchers during the AI Winter, the field experienced a revival in the 1990s and early 2000s. AI Researchers shifted their focus from Symbolic AI and rule-based systems to statistical and probabilistic approaches that are more promising. The emergence of machine learning algorithms, such as Support Vector Machines and Bayesian networks, enabled AI systems to demonstrate tangible successes in various applications. Machine learning systems have evolved dramatically over the years, progressing from basic models capable of simple tasks to complex neural networks that drive cutting-edge AI applications. Throughout this evolution, AI and ML have transitioned from rule-based systems to learning-based approaches, greatly expanding their capabilities and applications. In this keynote address, we will present an overview of their journey from inception to recent advances.

Keywords: Artificial Intelligence · Machine Learning · Symbolic System · Complex System

Attention Mechanism for Abnormal Behavior Detection from Video Surveillance

Vinothina V.

Kristu Jayanti College (Autonomous), Bengaluru, India

In todays fast-paced environment, detecting abnormal behaviour in real time is increasingly challenging, yet video monitoring remains vital for public safety. Traditional surveillance systems still rely heavily on manual observation, which is not only time-consuming but also prone to errors, especially when managing large volumes of footage. While machine learning (ML) algorithms have been widely used for behaviour detection, they often struggle with the complexity of unstructured video data. These algorithms typically require vast amounts of labelled data and can falter when dealing with imbalanced datasets, reducing their accuracy in identifying rare, yet critical, behaviours. Moreover, ML techniques tend to have slower learning rates, limiting their effectiveness in real-time scenarios and large-scale data applications. In contrast, deep learning models, especially convolutional neural networks (CNNs), have proven more effective by automatically learning feature representations directly from raw video data. However, CNNs still face challenges when it comes to capturing the temporal dynamics of behaviour across video frames. To address these challenges, the proposed method integrates a hybrid model that combines the strengths of CNNs and transformers with a context-based attention mechanism.

Modelled after the human brain's ability to focus on the most pertinent aspects of a scene, attention mechanisms integrated into deep learning frameworks have significantly improved both the accuracy and efficiency of abnormal behaviour detection. When combined with CNNs, these attention mechanisms emphasize key spatio-temporal features. Spatial attention allows the model to detect critical body movements and gestures, while temporal attention tracks changes in behaviour over time, identifying sudden or repetitive actions. This approach allows for real-time tracking of individual actions within the video frame, enabling timely alerts and interventions when abnormal behaviours are detected. When evaluated against existing methods, the proposed solution demonstrated superior performance, effectively overcoming the limitations associated with traditional machine learning approaches in behaviour recognition.

Keywords: Attention Mechanism · Abnormal Behaviour Detection · Video Surveillance · Deep Learning · Spatial Attention · Temporal Attention · Contextual Information

Zero Trust and Its Impact

Chander Shekhar Sharma

AUNZ - Cybersecurity Unit, Australia

The Zero Trust security model represents a fundamental shift in how organizations approach cybersecurity. Unlike traditional security frameworks that rely on a strong perimeter to protect internal networks, Zero Trust operates on the principle of "never trust, always verify." This approach assumes that threats can exist both inside and outside the network, and therefore, every access request—whether from a remote employee or an internal user—must be authenticated and authorized. By focusing on the identity of users and devices rather than the network location, Zero Trust transforms the way organizations secure their resources. At the heart of Zero Trust is the concept of identity as the new security perimeter. This means that organizations must implement robust identity verification mechanisms to ensure that only authorized individuals can access sensitive information. Multi-factor authentication (MFA) is a critical component of this strategy, requiring users to provide multiple forms of identification before gaining access. Additionally, risk-based adaptive authentication can further enhance security by adjusting access levels based on various factors, such as the user's location, device, and behaviour patterns. This continuous verification process helps organizations mitigate the risk of unauthorized access and data breaches.

The benefits of adopting a Zero Trust identity strategy are substantial. For instance, organizations can significantly reduce the likelihood of data breaches by ensuring that every access request is scrutinized. This not only protects sensitive information but also fosters a culture of security awareness among employees. Furthermore, Zero Trust can improve the user experience by streamlining access to resources while maintaining stringent security measures. For example, remote workers can securely access company applications from any location without compromising security, thereby enabling greater flexibility and productivity.

Use cases for Zero Trust identity are increasingly relevant in today's digital landscape. Organizations that have adopted cloud services or remote work policies can leverage Zero Trust principles to secure their environments effectively. For instance, a financial institution might implement Zero Trust to protect customer data by requiring MFA for all transactions and continuously monitoring user behaviour for anomalies. Similarly, a healthcare provider could use Zero Trust to ensure that only authorized personnel can access patient records, thereby complying with regulations and safeguarding sensitive information. By humanizing the approach to cybersecurity, Zero Trust not only addresses technical challenges but also aligns with the evolving needs of organizations and their employees in a rapidly changing digital world.

Empowering Academic Advising Through Smart Computing and Predictive Analysis

Ahmed Al-Brashdi

UTAS, Oman

The role of academic advising has evolved from static and transactional to proactive support, emphasizing innovative approaches to address the variations of students' performance and needs. By leveraging historical data on student performance, course complexity, and former evaluation, predictive analytics can identify students at risk of academic difficulty. An early warning system with comprehensive student data and predictive analysis based on historical data enables timely interventions, such as academic coaching or tutoring, to improve student retention and success. Moreover, using predictive analytics and AI will optimize resource allocation by balancing advisor workloads, and optimizing course scheduling. This data-driven approach contributes to cost-efficiency and improved student satisfaction. This approach can personalize the academic experience for each individual student by recommending tailored academic plans based on individual student strengths, goals, and performance patterns without overloading advisors. This approach fosters student engagement and accelerates progress towards degree completion. In UTAS – Oman, an in-house system enables advisors and students to effectively and efficiently engage with personalized study plans and course scheduling with emphasis on on-probation students. It shows improvements in student satisfaction and resource allocation.

Keywords: Academic advising · Predictive analysis · Smart computing · AI

Best Practices of Research Ethics

Vijay Singh Rathore

Shree K. Karni Universe College, Jaipur

Research ethics constitute a cornerstone of scientific inquiry, ensuring the credibility, integrity, and societal relevance of scholarly endeavours. This study delineates the fundamental principles governing ethical research practices, emphasizing adherence to honesty, objectivity, and transparency in all stages of research, from data collection to dissemination. Key ethical mandates include the avoidance of fabrication, falsification, and plagiarism, as well as the active protection of confidentiality, intellectual property rights, and the dignity of research participants.

The role of supervisors is highlighted as pivotal in fostering an ethical research environment, ensuring adherence to legal and institutional norms, and mentoring researchers in responsible practices. Equally, researchers are tasked with diligent participation in their projects, rigorous compliance with safety protocols, and acknowledgment of collaborative contributions.

Ethical challenges, such as authorship disputes, data ownership, and conflicts of interest, are addressed through structured protocols and mutual respect among collaborators. Furthermore, responsible dissemination of findings underscores the importance of accuracy and societal impact, avoiding premature or misleading claims. This framework underscores the collective responsibility of the research community to uphold ethical standards, thereby advancing scientific progress while maintaining public trust.

Contents – Part III

Artificial Intelligence and Machine Learning

Performance Evaluation of Machine Learning Algorithms for DDoS Attack Detection: A Comparative Study 3
Satyam Agrawal, Anup Ingle, Sagar Sudhakar Vanarase, Srinivas Chippalkatti, Nikhil Mohan Shinde, and Arati V. Deshpande

GeoVigilance Labs: An AI-Enabled System for Automated Land Use Monitoring 17
Harsh K. Dave, Bravim K. Purohit, Akshit A. Joshi, Devkumar R. Joshi, Swayam R. Pandya, Hetashriben D. Kansariwala, and Kamal Sutaria

Underwater Image Classification with Sloth-Enhanced Contrast Limited Adaptive Histogram Equalization (S-CLAHE) and Deep Learning Algorithms 26
P. Saravanan and K. Vadivazhagan

A Review of Deep-Learning-Based Models for Afaan Oromo Fake News Detection on Social Media Networks 39
Kedir Lemma Arega, Kula Kekeba Tune, Asrat Mulatu Beyene, Wegderes Tariku, and Nurhussen Menza Bune

Integrating AI and Block Chain for Enhanced Food Traceability and Safety in Restaurants: A Transformative Approach 59
Sachin Yele and Ratnesh Litoriya

PIRNet: Two-Step Deep Neural Network for Segmentation of Brain MRI with Efficient Loss Functions 71
Ashis Datta, Kunal Purkayastha, Palash Ghosal, and Rustom Ali Ahmed

Neural Networks Approach for Liver Cancer Classification 90
Vijaya Lakshmi Adluri, Mohan Aditya Dondapati, Maadesh Darisi, John Paul Devapooja, and Kaushik Chintalapati

Optimizing Tomato Leaf Disease Identification Using a Hybrid Spatial-Temporal Model and Attention Mechanism 99
Sajeev Ram Arumugam, P. Sheela Gowr, E. Anna Devi, J. Elavarasi, and Sankar Ganesh Karuppasamy

Machine Learning Strategies for Proactive Malware Detection 111
Hardik Mahendrabhai Patel and Shyam Kalariya

Generative AI Tool in Learning an Introductory Programming: A Research Perspective 120
M. Sivasakthi and A. Meenakshi

Evaluating the Effectiveness of Machine Learning Algorithms in Predicting Plant Diseases: A Comparative Analysis 133
Ashish K. Patil and Deepak Kumar Yadav

FLEATM: Fuzzy Logic-Based Energy-Aware Trust Based Routing in MANETs 144
R. Sivaranjani, R. Shankar, and S. Duraisamy

Smart Road Sensing: A Machine Learning Approach for Pothole Detection and Classification 160
Neel H. Dholakia, Vipul Ladva, Madhu Shukla, Nishant Kothari, Uvesh Sipai, Akshay Ranpariya, and Simrin Syed

From Threads to Algorithms: AI Innovations in Textile Production 173
Purvi Prajapati, Nirav Bhatt, Nikita Bhatt, Malav Champaneria, and Dhyani Panchal

Robust Dataset of Digital Handwritten Devanagari Script Images and Implementing Deep Learning for Handwritten Devanagari Character Recognition 187
Sanjay E. Pate and Rakesh J. Ramteke

Racism Detecting in Twitter Comments Using Metaheuristics Optimized Text Mining and Classification 217
Dobrivoje Dubljanin, Luka Jovanovic, Nebojsa Bacanin, Petar Spalevic, Branislav Radomirovic, Miodrag Zivkovic, and Angelina Njegus

Lower Limb Prosthetics Activity Recognition Using Optimized CNN-LSTM Model 232
M. Jeyasudha, S. Prakash, and G. Prakash

A Hybrid Approach to Cardiovascular Disease Detection: The Synergy of YOLOv9, CapsNet, and XGBoost 256
Gorapalli Srinivasa Rao and G. Muneeswari

Event Detection and Analysis from Social Media Data Using N-gram and Distil-BERT Model 266
Dharmendra Mangal and Hemant Makwana

Predictive Modeling of Genetic Diseases: A Comparison of Various Machine Learning Approaches 277
Rimjhim Sinha and Pradnya V. Kulkarni

Extracting the Pneumonia Signature: A Deep Learning Framework for Definitive Diagnosis 294
Malay Vyas and Apurva. A. Mehta

Leveraging Cloud Resources for Machine Learning-Based Spam Detection API 305
Ashish Revar, Bhavin Bhesaniya, and Kalpesh Wandra

From Pen to Pixel: Understanding Learning Shifts with Student Insights in E-Education 326
Kavita Yagyanarayan Jhanwar and Priti Sadaria

Navigating the Path: Exploring Career Choice Dynamics Among Secondary and Higher Secondary School Students 339
Falguni Mukeshbhai Barad and Amit Khjimjibhai Patel

Harassment Detection in Online Discourse Using Modified Metaheuristic Natural Language Processing 350
Stanislava Kozakijevic, Luka Jovanovic, Jasmina Perisic, Nebojsa Bacanin, Milos Mravik, Miodrag Zivkovic, and Milos Antonijevic

Analyzing Extractive Summarization Methods for Kannada Using Sentence Ranking and K-Means Algorithm 364
Dakshayani Ijeri, Pushpa B. Patil, and Sanjana Mirajkar

Real-Time Facial Emotion Recognition Through Neural Networks 379
Dhairya A. Bhatt, Harshil Rajpara, Anjali Diwan, and Rajesh Mahadeva

Deepfake Detection Using ResNet50: Performance Analysis on Celeb-DF Dataset 397
Yashoda Alpesh Chouhan, Chetankumar Chudasama, and Deepak Kumar Verma

Performance of Linear and Polynomial Kernels of SVM Towards Kidney Cyst Detection 405
Rahul Nimbai, Mayur Patil, Veereshkumar Rathod, Santosh Pattar, and Prema T. Akkasaligar

Author Index 417

Artificial Intelligence and Machine Learning



Performance Evaluation of Machine Learning Algorithms for DDoS Attack Detection: A Comparative Study

Satyam Agrawal[1(✉)], Anup Ingle[1], Sagar Sudhakar Vanarase[2], Srinivas Chippalkatti[3], Nikhil Mohan Shinde[4], and Arati V. Deshpande[5]

[1] Department of Electronics and Telecommunication, Vishwakarma Institute of Information Technology, Pune, India
{satyam.22010754,anup.ingale}@viit.ac.in
[2] Department of Electronics and Telecommunication, Cummins College of Engineering for Women, Pune, India
sagar.vanarase@cumminscollege.in
[3] Department of Engineering Sciences, Vishwakarma University, Pune, India
srinivas.chippalkatti@vupune.ac.in
[4] Department of Mechanical Engineering, Vishwakarma University, Pune, India
nikhil.shinde@vupune.ac.in
[5] Department of Computer Engineering, BRACT'S Vishwakarma Institute of Technology, Pune, India
arati.deshpande1@vit.edu

Abstract. The proliferation of Distributed Denial of Service (DDoS) attacks poses a significant threat to internet users and organizations. This paper presents a comprehensive evaluation of four machine learning algorithms, Catboost, Logistic Regression, Gradient Boost, and Naïve Bayes aimed at effectively detecting and classifying benign and attack transactions. Utilizing four open-source datasets, including CICDDOS19 (Portmapper attack dataset), DDOS Backscatter TCP dataset, DDOS Backscatter ICMP attack dataset, and the SDN dataset sourced from publicly available repositories, meticulous preprocessing techniques were employed to clean and scale the datasets before training the machine learning models. The performance evaluation of the algorithms was conducted based on key metrics such as Accuracy, F1 score, Precision, and Recall. The comparative analysis presented in this paper facilitates the determination of the best-performing algorithm for DDoS attack detection across diverse datasets, thereby offering valuable insights for enhancing cybersecurity measures.

Keywords: Machine learning · DDOS Attacks · Comparative Analysis · Cybersecurity · Detection

1 Introduction

With the rapid advancement and widespread adoption of technology, there has been a corresponding increase in networking attacks. India saw a notable 15% surge in cyberattack incidents, positioning it as the second most targeted nation in 2023 [20]. In essence,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 3–16, 2025.
https://doi.org/10.1007/978-3-031-86296-0_1

a networking attack entails any unauthorized activity perpetrated on a system [19]. These attacks exploit various vulnerabilities in networking protocols or human errors to gain illicit access, pilfer confidential data, or disrupt system operations, rendering it incapable of servicing legitimate requests [19]. Common forms of networking attacks include Distributed Denial of Service (DDoS) attacks, wherein a multitude of botnets inundate a victim's server with an overwhelming volume of service requests, hindering its ability to process genuine user requests [1, 21–23]. Another common attack vector is the Man-in-the-Middle attack, in which an attacker intercepts and possibly modifies communication between a user and a service provider to steal sensitive data [1]. Phishing is yet another tactic employed by attackers to deceive victims into divulging personal information by clicking on malicious links [1]. However, our primary focus lies on DDoS attacks, which pose a significant threat to the availability and integrity of services. DDoS attacks not only disrupt the targeted system but also jeopardize other connected devices, such as IoT devices [18]. The ramifications of such attacks are profound, especially for service-oriented enterprises like e-commerce websites and tech giants. Beyond financial losses, these attacks inflict reputational damage, impairing long-term performance. Moreover, resource wastage further compounds the detrimental effects, while data breaches during these incidents expose enterprises to legal repercussions [18]. To mitigate such threats, two key systems come into play: The Intrusion Detection System (IDS) and The Intrusion Prevention System (IPS) [17, 24]. IDS functions by monitoring network traffic, analyzing data against predefined attack patterns, and alerting network administrators upon detection of suspicious activity [17]. Conversely, IPS not only detects but also actively blocks malicious packets, thwarting potential attacks before they infiltrate the system [3].

2 Related Work

In addressing security challenges within smart grid systems, [4] explores intrusion detection leveraging machine learning models. This study delves into various attack scenarios and utilizes the CIC-IDS2018 dataset for research purposes. Evaluating four classification algorithms—Random Forest, Naive Bayes, Support Vector Machine, and Extreme Gradient Boosting—the author employs feature selection techniques such as Boruta and Recursive Feature Elimination. Notably, Extreme Gradient Boosting and Random Forest emerge as top performers, boasting high recall rates. The paper concludes by advocating for further enhancements and exploration of alternative classifiers to bolster intrusion detection within smart grid systems. Meanwhile, [5] proposes a DDoS attack detection system integrating the Gradient Boosting classification algorithm (GBT) with the Apache Spark to avoid the complexities of identifying sophisticated DDoS attacks. By combining machine learning with parallel computing, the framework aims to achieve practical performance rates while minimizing detection delays. Experimental evaluations on real DDoS datasets demonstrate the system's effectiveness across metrics such as accuracy, recall, precision, and false positive rate. Furthermore, the study investigates the impact of dataset size and algorithm parameters on processing delays, highlighting the efficient DDoS attack detection facilitated by GBT integrated with Spark. In a similar vein, [6] discusses various algorithms for attacks in enterprise the networks using machine learning methods. Evaluating various classifiers on the CICIDS2017 dataset for double and

multiclass classification of attacks, consisting DoS/DDoS, Port Scan, Brute Force, Web Attack, Bot, and Infiltration, Catboost and Light GBM emerge as top performers. Despite Catboost slightly outperforming Light GBM, both algorithms demonstrate high accuracy, F1-score, precision, and recall. However, the author notes that Catboost requires longer training time compared to Light GBM, emphasizing the importance of efficient attack detection in enterprise networks, and suggesting the integration of machine learning with signature based methods for enhanced security. Whereas, [7] presents a machine learning based approach for detection of Distributed Denial of Service (DDoS) attack and preventing botnet formation. Utilizing various algorithms on the NSL KDD dataset, the author proposes a hybrid heterogeneous multi-classifier ensemble learning system for robust detection and prevention. Feature selection and preprocessing techniques are employed for model training and evaluation, with logistic regression achieving an accuracy of 96.35% in distinguishing between legitimate and malicious URLs. This paper highlights the critical role of machine learning in mitigating evolving cyber threats. In [8], the author evaluates classification algorithms for detection of the DDoS attacks utilizing CICDDoS2019 dataset. Six supervised algorithms were scrutinized, including the Logistic Regression, Naïve Bayes, SVM, Decision Tree, Random Forest, and K-NN. Notably, tree-based, and distancebased classifiers emerged as top performers. While unbalanced datasets exhibited high accuracy, balanced datasets demonstrated superior results, notably with the Decision Tree, Random Forest, and K-NN achieving nearly perfect accuracy rates. The study underscores the importance of exploring larger datasets and diverse attack types to bolster detection accuracy. Meanwhile, [9] delves into identifying DDoS attacks in the various Internet of Things (IoT) network devices through machine learning methods. The author proposes an approach that enriches feature sets with a hashing algorithm for various device configuration files. Binary and multiclass classifications were executed using Gradient Boosting, AdaBoost, and Catboost on the CICDDoS2019 dataset. Catboost exhibited outstanding results in binary classification, achieving 99.3% accuracy, while also outperforming others in multiclass classification with 97% accuracy. Feature importance analysis led to an 11.5% reduction in computing resource consumption. The study concludes that while reducing feature sets slightly affected classification accuracy, it significantly enhanced computational efficiency.

3 Dataset and Data Cleaning

3.1 Primarily focusing on the CICDDOS2019 (Portmap Dataset) [12], sourced from the Canadian Institute for Cybersecurity (CIC). However, during initial data exploration and visualization, several limitations were identified and addressed through meticulous data cleaning processes:

- Imbalanced Dataset: The dataset exhibited a significant class imbalance, with a vast majority of entries representing attack scenarios. To mitigate this issue, suitable undersampling techniques were applied to balance the dataset.
- Handling Null Values: A single null value was detected in the “Flow Bytes/s” column, which was subsequently dropped from the dataset to ensure data integrity.

- Data Type Conversion: Certain columns such as “Flow ID”, “Source IP”, “Destination IP”, “Timestamp”, and “Simillar HTTP” were in object datatype, unsuitable for direct utilization in machine learning algorithms. Consequently, “Flow ID” and “Timestamp” columns were dropped, while “Simillar HTTP” was removed due to its high correlation with the label column. However, “Source IP” and “Destination IP” were deemed significant features and were converted using label encoding to numeric values for model compatibility.
- Handling Negative Values: Several columns contained negative values, rendering them incompatible with machine learning models. Accordingly, columns such as “Init_Win_bytes_forward” and “Init_Win_bytes_backward” were dropped due to the prevalence of negative values. Negative values from other columns like “Fwd Header Length” and “min_seg_size_forward” were also removed.
- Handling Infinite Values: “Flow Bytes/s” and “Flow Packets/s” columns contained infinite values, posing limitations on data processing. As replacing these values would introduce bias, they were consequently dropped from the dataset. By addressing these limitations through rigorous data cleaning and preprocessing techniques, the dataset was refined and made suitable for subsequent analysis and model training.

3.2 The DDoS backscatter traffic dataset [11] comprises data features generated by victims of DDoS attacks during an ongoing assault. These features involve packets, known as backscatter traffic, which the victim generates in response to query packets with spoofed source IP addresses sent by the attacker. The considered dataset is divided into two distinct CSV files: icmp.csv and tcp.csv

icmp.csv: This dataset pertains to annotated and anonymized backscatter data for the ICMP protocol, encompassing 34 columns as features and 1 column named “DDOS,” denoting whether the transaction is classified as an attack (“True”) or benign (“False”). During data visualization, it was noted that the dataset consisted of 51,507 entries, with 7,004 labelled as “True” and the remainder as “False,” indicating an imbalance. To address this, appropriate undersampling algorithms were employed. Additionally, columns such as “DATE_START” and “DATE_END” were found to be of object datatype and were subsequently dropped due to redundancy, with the “DURATION” column providing similar information. The “SRC_IP” column was encoded using label encoding to convert it into numeric values for model compatibility.

tcp.csv: Similarly, this dataset represents annotated and anonymized backscatter data for the TCP protocol, comprising 36 columns as features and 1 column denoting the transaction as either an attack or benign. While facing similar limitations as the icmp.csv dataset, including an unbalanced distribution of entries, comparable actions were taken during data visualization and preprocessing stages to ensure dataset integrity and model compatibility. By addressing these limitations and employing appropriate data preprocessing techniques, the datasets were refined and prepared for subsequent analysis and model training.

3.3 DDOS Attack SDN-Dataset [10] is a dataset which is specific to Software Defined Networking (SDN) and was formed using a emulator known as the Mininet emulator. It consists of a total of 23 features, of which some are extracted from the switches while others features are generated by calculations. The dataset includes a column labelled “label,” which determines whether a transaction is benign or an attack. With 104,345 entries

in total, the dataset contains 63,561 benign entries and 40,784 attack entries, maintaining a balanced distribution and negating the need for under-sampling or over-sampling techniques. During data visualization and preprocessing stages, several limitations were identified and addressed:

- Handling Null Values: The columns "rx_kbps" and "tx_kbps" each contained 506 null values, which were subsequently dropped from the dataset to maintain data integrity.
- Data Type Conversion: Columns such as "src," "dst," and "Protocol" were of object datatype. "src" and "dst," representing addresses, were encoded into numeric format using label encoding. However, the "Protocol" column was dropped from the dataset due to the unbalanced distribution of attack entries among different protocols, potentially causing errors in detection.
- Handling Negative Values: Columns "pktperflow," "byteperflow," and "pktrate" contained negative values, which were deemed incompatible with the dataset. Consequently, these columns were dropped to ensure model compatibility and accuracy.

By addressing these limitations and employing appropriate data preprocessing techniques, the DDOS Attack SDN Dataset was refined and prepared for subsequent analysis and model training.

4 Methodology

4.1 Feature Scaling

After data visualization in excel it was found that values present in the features of our dataset are of varying range. To bring all the values of our dataset in similar range we have performed the MaxAbsScaling Technique of the python's scikit learn. The MaxAbsScaling technique works by dividing each and every entry with its maximum possible entry from the data. This keeps the values in the range of $[-1,1]$ which makes it easier for our machine learning model to process the data as compared to the huge values. We have performed feature scaling as when necessary according to the machine learning model used.

4.2 Feature Selection

We have used scikit learns Selectkbest feature algorithm for performing feature selection on our datasets. Selectkbest algorithm is a feature selection technique used in machine learning to identify the top k most relevant features for predicting the target variable. It works by applying statistical scoring functions to each feature, such as chi-squared, mutual information, or F-score (chi-squared in our case), to quantify its relationship with the target. Following this, the Features are ranked according to their scores, and the top k features with the highest rankings are chosen for inclusion in the final dataset. This helps in identifying the best features from our dataset which affect the target column the most and removing redundant features. We have used varying values of k with each and every dataset in order to ensure best performance metrics.

4.3 Algorithms Used and Their Generalized Computational Efficiencies

After applying the Selectkbest feature algorithm the selected k features were passed to the respective machine learning algorithm.

Logistic Regression

The Logistic Regression [2] is said to be a supervised machine learning algorithm which is used for achieving outputs in classification tasks. The sole goal of using the algorithm is to predict the probability that whether the given instance belongs to a given class or not. Since the logistic regression is mainly applicable for binary classification tasks, its desired output is in discrete form, such as true/false, yes/no, 0/1. But as it is a probability-based function it produces real valued output between 0 and 1. So as to obtain discrete values, the concept of thresholding in together with the sigmoid function is used [2]. Sigmoid function maps real values to a range between 0 and 1, and applying a threshold allows us to obtain the output in the desired format. For example, let us assume α (is a number between 0 and 1) as threshold, so any time when the output is achieved it is compared with α, if it is less than α the output maybe predicted as false and true in case the where the obtained value is greater than α. The actual value of α is determined during the training of the algorithm on our dataset. The primary benefits of logistic regression include it is easier to implement and interpret due to which it requires less computational power and it is also suitable to implement on linearly separable dataset, linearly separable dataset is a graph in which a straight line divides the two distinct classes of data.

Naïve Bayes

The Naive Bayes [1] is considered a probability based supervised machine learning algorithm that is commonly used for classification tasks. Naive Bayes relies on the strong and naive assumption that all features in a dataset are independent of each other when conditioned on the class label. Due to the assumption, Naive Bayes often fails to perform well in actual real world problem statements [1]. During its training phase, The Naive Bayes mathematically calculates the prior probability of every class label and likelihood of every feature present in each class label. This involves counting occurrences of features within every class. Whereas, while making new predictions for new data, Naive Bayes mathematically calculates the posterior probability of every class label given the observed features using Bayes' theorem. It then selects class label with the highest posterior probability as the predicted class.

Gradient Boosting

The Gradient Boosting is an ensemble learning method which used for both classification and regression tasks. In its workings, it sequentially constructs a ensemble of weak learners, typically decision trees as shown in Fig. 1 aiming to minimize errors iteratively. The process begins with initializing the model, usually with a simple prediction of the target variable. Subsequently, it computes pseudo-residuals, representing the errors unaccounted for by the current model. Then, it fits a weak learner to these residuals, focusing on minimizing them. The model is updated with the new learner's predictions, and the process repeats for a defined number of iterations, refining predictions with each step. The ultimate prediction is derived from totaling the predictions of

all the weak learners. This iterative refinement strategy enables Gradient Boosting to produce accurate predictions by continuously addressing the model's deficiencies.

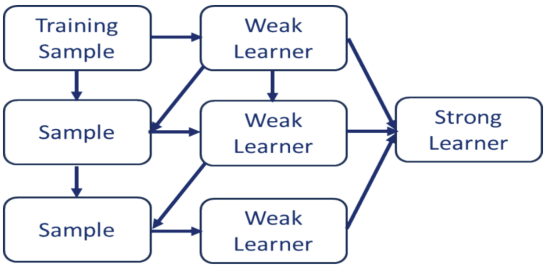


Fig. 1. Depicts the concept of gradient boosting algorithm [16]

Cat Boosting

The Catboost [3] is a machine learning algorithm which is developed by Yandex that is specifically designed to handle categorical features efficiently. It stands for "Categorical Boosting" and is relies on the gradient boosting decision trees. Catboost builds an ensemble of decision trees in a sequential order. Every tree is trained to correct the inaccuracy made by the preceding ones, with the ultimate prediction outcome being the sum of predictions from all trees. Catboost optimizes the loss function by incrementally adding new decision trees to the ensemble during each iteration [3]. Each tree is trained on the gradient of the loss function concerning the predictions made by the preceding trees. The loss function used in our case is the 'Logloss' as we have binary target variable. Apart from handling the categorical features, Catboost is also more efficient due to its inbuilt data preprocessing techniques like handling missing values and automatic feature scaling. Additionally, CatBoost implements techniques such as robust tree boosting, ordered boosting, and random permutations as shown in block diagram in Fig. 2 for feature combinations which makes it robust towards overfitting.

Considering the datasheets of each algorithm we can conclude the following as the generalized computational efficiency order from highest to lowest:

- Naïve Bayes (Most Efficient)
- Logistic Regression
- CatBoost
- Gradient Boosting (Least Efficient)

5 Implementation

The research implementation involved the utilization of several libraries for diverse purposes. The Scikit-learn library was instrumental in executing feature scaling, feature selection, implementing algorithms, and performing performance evaluation tasks. Data handling tasks, on the other hand, were facilitated by the Panda's library, which allowed for efficient manipulation of data in the form of data frames. Various mathematical operations were carried out using the NumPy library, ensuring streamlined numerical

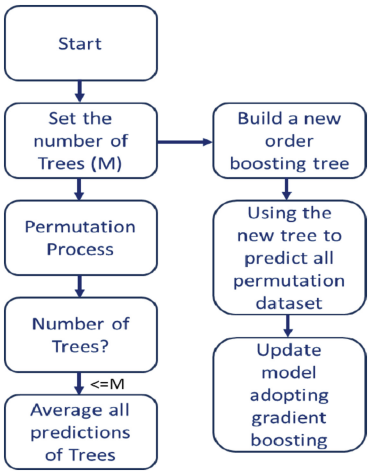


Fig. 2. Depicts Working of CatBoost algorithm [15]

computations. The training and testing phases of the research were conducted by a machine having the following specifications:

- Processor: AMD Ryzen 7 5800HS having Radeon Graphics, operating at a frequency of 3.20 GHz
- Installed RAM: 8.00 GB out of which 7.41 GB is usable.
- System type: 64-bit operating system, x64-based processor

These hardware specifications provided a robust computing environment for carrying out the experiments and analyses outlined in the research.

6 Mathematical Modelling

Actual implementation of the algorithms involves the Mathematical equations. Following are the algorithm and their respective equations-

6.1 Logistic Regression

Consider X as input containing set of input features $x1$, $x2$, xn,

Thus, we can derive a linear relation z as,

$$\mathrm{z} = \beta 0 + \beta 1\mathrm{x}1 + \beta 2\mathrm{x}2 + \ldots + \beta \mathrm{nxn} \tag{1}$$

where,

β0 is considered as the intercept term,

β1, β2,…, βn are considered to be the coefficients associated with each features $x1$, $x2$,…, xnx1, x2,…, xn respectively.

The equation for the logistic function (sigmoid) is given as,

$$\mathrm{p(x)} = \frac{1}{1 + e^{(-z)}} \tag{2}$$

where,

p(x) is considered as the predicted probability that the given input x turns out as a member to the positive class.

e, the base of the natural logarithm (Euler's number)

The Final equation for model becomes,

$$\begin{aligned} \text{y} = 1 \text{ if, } \text{p(x)} &> \text{threshold} \\ \text{y} = 0 \text{ if, } \text{p(x)} &< \text{threshold} \end{aligned} \tag{3}$$

The threshold is the p(x) value obtained from training data.

6.2 Naïve Bayes Algorithm

The Naïve bayes algorithm is a machine learning algorithm based on the mathematical bayes theorem due to which its equation turns out to be same as the bayes theorem which is given by

$$P(y|X) = \frac{\text{P(X}|y) \cdot \text{P(y)}}{\text{P(X)}} \tag{4}$$

P(y|X) is considered to be the posterior probability of class y given the features X.

$P(X|y)$ is considered to be the likelihood of observing the features X given class y.

$P(y)$ is considered to be the prior probability of class y.

$P(X)$ is the considered to be probability of observing the features X across all classes.

6.3 Gradient Boosting

Gradient Boosting involves minimizing a loss function $L(y,F(x))$ where y represents the true labels and F(x) represents the current prediction.

The objective of Gradient Boosting is to find the ensemble of weak learners (typically decision trees) that minimizes the loss function. This is often represented as:

$$\sum_{i=1}^{n} L(y_i, F(x_i)) + \sum_{k=1}^{K} \Omega(f_k) \tag{5}$$

Where,

K is the number of weak learners (trees), fk represents each individual tree, and $\Omega(fk)$ is a regularization term to prevent overfitting.

At each iteration m, the gradient of the loss function with respect to the current prediction $F(x)$ is computed for each sample i

$$\text{g}_{im} = \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \tag{6}$$

The negative gradient of the loss function is used as the pseudo-residuals, representing the errors unaccounted for by the current model

$$\mathbf{g}_{im} = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \tag{7}$$

Each weak learner is trained on the pseudo-residuals with the goal of reducing the loss function. The final output of each tree is denoted as $fm(x)$.

Gradient Boosting typically employs a learning rate (η) which scales the contribution of each tree. This helps to prevent overfitting and allows for better generalization.

The predictions are updated at each iteration by adding the output of the new weak learner, scaled by the learning rate which are given by,

$$F(x) \leftarrow F(x) + \eta . f_m(x)$$

7 Result

Table 1 summarizes the results obtained from applying the discussed algorithms to the datasets. Performance metrics such as Accuracy, F1-score, Precision, and Recall were utilized to evaluate algorithm performance. The order of performance based on accuracy in decreasing order for each dataset is as follows:

7.1 CICDDOS2019 Portmap Dataset

- Gradient Boosting → Cat Boost → Logistic Regression → Naïve Bayes
- Gradient Boosting Algorithm: 99.84%

7.2 TCP Dataset

- Cat Boost → Gradient Boosting → Logistic Regression → Naïve Bayes
- CatBoost Algorithm: 99.42%

7.3 ICMP Dataset

- Cat Boost → Gradient Boosting → Logistic Regression → Naïve Bayes
- CatBoost Algorithm: 99.84%

7.4 SDN Dataset

- Cat Boost → Gradient Boosting → Logistic Regression → Naïve Bayes
- CatBoost Algorithm: 99.74%

These rankings highlight the effectiveness of specific algorithms in accurately classifying DDoS attacks across the different datasets.

Based on the results obtained and depicted in Fig. 3, it is evident that across various datasets, specific algorithms outperform others in terms of evaluation metrics. For instance, in the CICDDOS2019 Portmap Dataset, the Gradient Boosting Algorithm demonstrates superior performance with an accuracy of 99.84%. Conversely, in the TCP dataset, the Catboost algorithm emerges as the top performer, achieving an accuracy of 99.42%. Similarly, in the ICMP and SDN datasets, Catboost stands out as the best-performing algorithm, boasting accuracies of 99.84% and 99.74%, respectively. These findings underscore the effectiveness of specific algorithms in differentiating and classifying DDoS attacks across diverse datasets.

Table 1. Performance metrics of algorithms in various dataset

Dataset	Algorithm	Feature Scaling (Y/N)	K value	Accuracy	F1-score	Precision	Recall
CICDDOS2019 Portmap	CB	N	25	0.9979	0.9984	0.9992	0.9976
	LR	N	25	0.9954	0.9964	0.9984	0.9944
	GB	N	25	0.9984	0.9988	1.0	0.9976
	NB	N	5	0.7270	0.8233	0.7005	0.9984
TCP	CB	N	20	0.9942	0.9909	0.9925	0.9893
	LR	N	20	0.9816	0.9704	0.9784	0.9625
	GB	N	20	0.9941	0.9907	0.9930	0.9884
	NB	Y	5	0.9162	0.8548	0.9480	0.7782
ICMP Dataset	CB	N	10	0.9984	0.9976	0.9974	0.9979
	LR	N	10	0.9973	0.9960	0.9956	0.9964
	GB	N	10	0.9976	0.9964	0.9943	0.9985
	NB	N	10	0.8162	0.8212	0.8162	0.8162
SDN Dataset	CB	N	10	0.9974	0.9967	0.9942	0.9992
	LR	Y	17	0.7557	0.6708	0.7037	0.6409
	GB	N	15	0.9970	0.9961	0.9939	0.9983
	NB	Y	5	0.7332	0.6643	0.6495	0.6798

8 Conclusion

From Table 1 and Fig. 3, it's evident that Gradient Boosting and Catboost algorithms exhibit similar performance, with only a slight difference in output values. This similarity in performance may stem from their comparable training and testing procedures. Despite this, Logistic Regression, while underperforming compared to CB and GB algorithms, still demonstrates decent performance across three out of the four datasets. Its computational efficiency and ease of implementation make it a viable option for DDOS attack detection.

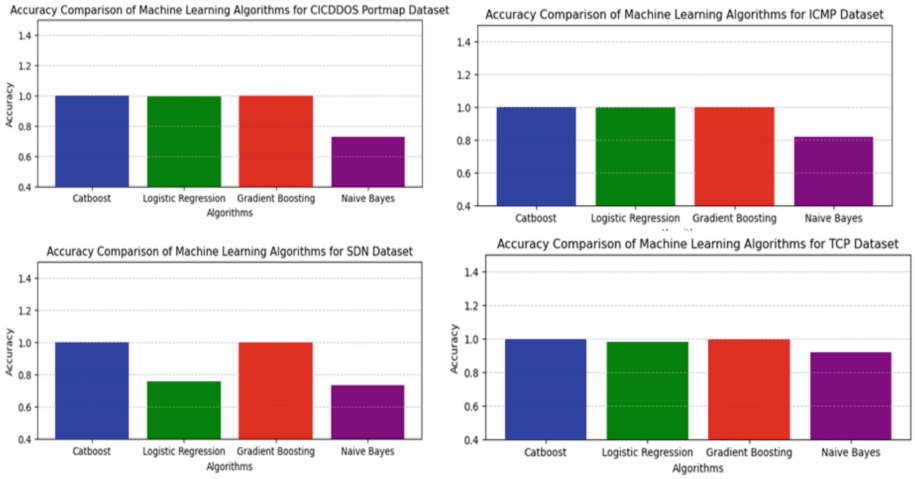


Fig. 3. Accuracy Plot of each dataset for all the algorithms

On the other hand, Naïve Bayes algorithm emerges as the worst-performing among all. Even with similar k values to other algorithms, it performs even worse, indicating a significant discrepancy. Moreover, it required feature scaling to produce relatively better outputs. This poor performance could be attributed to its basic assumption of feature independence, which does not hold true in our dataset.

In conclusion, tree-based algorithms significantly outperform simpler algorithms in detecting DDOS attacks. However, logistic regression, being simpler and computationally less expensive than tree-based algorithms, still yields decent performance in most cases.

References

1. Yang, F.-J.: An implementation of naive bayes classifier. In: 2018 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 301–306. Las Vegas, NV, USA (2018). https://doi.org/10.1109/CSCI46756.2018.00065
2. Bisong, E.: Logistic regression. In: Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, Berkeley, CA (2019). https://doi.org/10.1007/978-1-4842-4470-8_20
3. Dorogush, A.V., Ershov, V., Gulin, A.: CatBoost: gradient boosting with categorical features support. arXiv preprint arXiv:1810.11363 (2018)
4. Roy, D.D., Shin, D.: Network intrusion detection in smart grids for imbalanced attack types using machine learning models. In: 2019 International Conference on Information and Communication Technology Convergence (ICTC), pp. 576–581. Jeju, Korea (South) (2019). https://doi.org/10.1109/ICTC46691.2019.8939744
5. Alsirhani, A., Sampalli, S., Bodorik, P.: DDoS detection system: utilizing gradient boosting algorithm and apache spark. In: 2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE), pp. 1–6. Quebec, QC, Canada (2018). https://doi.org/10.1109/CCECE.2018.8447671

6. Bakhareva, N., Shukhman, A., Matveev, A., Polezhaev, P., Ushakov, Y., Legashev, l.: attack Detection in Enterprise Networks by Machine Learning Methods. In: 2019 International Russian Automation Conference (RusAutoCon), pp. 1–6. Sochi, Russia (2019). https://doi.org/10.1109/RUSAUTOCON.2019.8867696
7. Chavan, N., Kukreja, M., Jagwani, G., Nishad, N., Deb, N.: DDoS attack detection and botnet prevention using machine learning. In: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1159–1163. Coimbatore, India (2022). https://doi.org/10.1109/ICACCS54159.2022.9785247
8. Gohil, M., Kumar, S.: Evaluation of classification algorithms for distributed denial of service attack detection. In: 2020 IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 138–141. Laguna Hills, CA, USA (2020). https://doi.org/10.1109/AIKE48582.2020.00028
9. Parfenov, D., Kuznetsova, L., Yanishevskaya, N., Bolodurina, I., Zhigalov, A., Legashev, L.: Research application of ensemble machine learning methods to the problem of multiclass classification of ddos attacks identification. In: 2020 International Conference Engineering and Telecommunication (En&T), pp. 1–7. Dolgoprudny, Russia (2020). https://doi.org/10.1109/EnT50437.2020.9431255
10. Ahuja, N., Singal, G., Mukhopadhyay, D.: DDOS attack SDN Dataset. Mendeley Data **V1** (2020). https://doi.org/10.17632/jxpfjc64kr.1
11. Zadnik, M.: DDoS backscatter dataset. Mendeley Data **V1** (2022). https://doi.org/10.17632/37zz4pvjzp.1
12. Sharafaldin, I., Lashkari, A.H., Hakak, S., Ghorbani, A.A.: Developing realistic Distributed Denial of Service (DDoS) attack dataset and taxonomy. In: 2019 International Carnahan Conference on Security Technology (ICCST), pp. 1–8. Chennai, India (2019). https://doi.org/10.1109/CCST.2019.8888419
13. Pedregosa, F., et al.: Scikit-learn: machine Learning in Python
14. Richard, Z., Pitassi, T.: A gradient-based boosting algorithm for regression problems. Adv. Neural Inform.n Process. Syst. **13** (2000)
15. Yalçin, T., Solà, P., Stefanidou-Voziki, P., Dominguez-Garcia, J.L., Demirdelen, T.: Exploiting digitalization of solar PV plants using machine learning: digital twin concept for operation. Energies **16**, 5044 (2023). https://doi.org/10.3390/en16135044
16. Jia, D., Xue, R.: Research on earnings management of growth enterprise market in china stock market: comparative analysis based on the BPNN, GBDT, and MLR models. Comput. Intell. Neurosci. **2022**, 1–9 (2022). https://doi.org/10.1155/2022/6064536
17. Coulibaly, K.: An overview of Intrusion Detection and Prevention Systems (2020)
18. Tushir, B., Dalal, Y., Dezfouli, B., Liu, Y.: A quantitative study of DDoS and E-DDoS attacks on WiFi smart home devices. IEEE Internet Things J. **8**(8), 6282–6292 (2021). https://doi.org/10.1109/JIOT.2020.3026023
19. Top 10 Common Types of Network Security Attacks Explained. https://cisomag.com/top-10-common-types-of-network-security-attacks-explained/
20. India witnesses 15% rise in cyber attack cases in 2023; emerges as 2nd most targeted nation. https://www.livemint.com/news/india/india-witnesses-15-rise-in-cyber-attack-cases-in-2023-emerges-as-2nd-most-targeted-nation-11705939863447.html
21. Gupta, A., Pandey, O.J., Shukla, M., Dadhich, A., Mathur, S., Ingle, A.: Computational intelligence based intrusion detection systems for wireless communication and pervasive computing networks. In: 2013 IEEE International Conference on Computational Intelligence and Computing Research, pp. 1–7. Enathi, India (2013). https://doi.org/10.1109/ICCIC.2013.6724156
22. Gupta, A., Pandey, O.J., Shukla, M., Dadhich, Ingle, A., Ambhore, V.: Intelligent perpetual echo attack detection on user datagram protocol port 7 using ant colony optimization. In:

2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies, pp. 419–424. Nagpur, India (2014). https://doi.org/10.1109/ICESC.2014.82
23. Gupta, A., Pandey, O.J., Shukla, M., Dadhich, Ingle, A., Gawande, P.: Towards context-aware smart mechatronics networks: Integrating Swarm Intelligence and Ambient Intelligence. In: 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 64–69. Ghaziabad, India (2014). https://doi.org/10.1109/ICICICT.2014.6781254
24. Kolte, S., Ingle, A., Kshirsagar, K., Gawande, P.G., Kumbhar, K.A., Kawchale, R.S.; Traffic flow optimization using CNN and GCN for adaptive signal control. In: 2023 3rd Asian Conference on Innovation in Technology (ASIANCON), pp. 1–4. Ravet IN, India (2023). https://doi.org/10.1109/ASIANCON58793.2023.10269944

GeoVigilance Labs: An AI-Enabled System for Automated Land Use Monitoring

Harsh K. Dave(✉), Bravim K. Purohit, Akshit A. Joshi, Devkumar R. Joshi, Swayam R. Pandya, Hetashriben D. Kansariwala, and Kamal Sutaria

Parul University, Vadodara, India
harsh.dave1506@gmail.com

Abstract. GeoVigilance Labs is an innovative AI-driven system designed to combat unauthorized land use changes. Leveraging cutting-edge technology, this system enables streamlined data collection, real-time monitoring, and immediate notifications to relevant authorities. The research delineates the system's methodology, emphasizing its potential to significantly enhance regulatory enforcement, preserve environmental integrity, and foster improved land management practices. By empowering local authorities and integrating state-of-the-art technology, GeoVigilance Labs stands as a transformative solution for addressing unauthorized land use, offering a sustainable approach for governance and environmental sustainability.

Keywords: AI-driven monitoring · unauthorized land use · regulatory enforcement · environmental integrity · land management · governance

1 Introduction and Background

Unauthorized land use poses significant challenges globally, impacting environmental sustainability, regulatory compliance, and land management practices. Existing monitoring systems often face limitations in detecting and addressing these alterations promptly and comprehensively. To overcome these challenges, the development of an innovative AI-driven system, known as GeoVigilance Labs, has emerged as a solution.

GeoVigilance Labs aims to revolutionize land use monitoring by amalgamating cutting-edge technology with comprehensive data collection methodologies. This system leverages advanced AI algorithms and satellite imagery analysis to detect unauthorized land use changes in real-time. The integration of local authorities, notably Talatis and Gram Panchayats, ensures a bottom-up approach to monitoring, empowering grassroots governance.

The need for such a system arises from the increasing complexities in monitoring vast land areas and swiftly addressing unauthorized changes. Existing systems often lack the efficiency, speed, and comprehensive coverage required for effective regulatory enforcement and environmental preservation.

By utilizing AI for monitoring and empowering local governance structures, GeoVigilance Labs seeks to bridge these gaps and revolutionize land use management practices.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 17–25, 2025.
https://doi.org/10.1007/978-3-031-86296-0_2

The introduction of this innovative system holds promise in enhancing regulatory compliance, preserving environmental integrity, and fostering more efficient land management, thereby addressing crucial concerns in contemporary land use governance.

This paper delineates the methodology, results, implications, and future prospects of GeoVigilance Labs, underscoring its potential as a transformative solution in land use monitoring and governance.

2 Literature

Land use monitoring and the detection of unauthorized changes have been extensively studied in recent literature, focusing on technological advancements, methodologies, and governance structures relevant to systems akin to GeoVigilance Labs.

Remote Sensing Techniques for Land Use Monitoring: [1]
Studies have highlighted the efficacy of remote sensing methods, particularly utilizing satellite imagery, for monitoring changes in land use. These methodologies encompass spectral analysis and machine learning algorithms to detect alterations in land use patterns accurately.

AI Integration in Land Use Monitoring Systems: [2]
Research demonstrates the potential of integrating artificial intelligence (AI) in real-time detection and classification of land use changes. Utilizing AI algorithms such as YOLOv8 and CNN architectures, these studies showcase precise and efficient detection methodologies.

Community Engagement and Governance in Land Management: [3]
Involvement of local communities and governance structures significantly impacts effective land management. Studies emphasize the role of local authorities and community engagement in monitoring and enforcing land use regulations.

Scalability and Cloud-Based Solutions for Land Monitoring: [4]
Advancements in cloud computing play a crucial role in enhancing the scalability of land monitoring systems. Cloud-based infrastructures accommodate increased data loads and improve scalability in land monitoring processes.

Environmental Impact Assessment in Land Use Governance: [5]
Accurate land use monitoring remains critical in preserving environmentally sensitive areas. Studies stress the importance of technological innovations in assessing environmental impact and conservation efforts within land use governance frameworks.

This Literature Review encapsulates recent advancements in land use monitoring methodologies, encompassing remote sensing technologies, AI integration, community engagement, scalability through cloud-based solutions, and the pivotal role of effective governance in preserving ecological integrity. These studies serve as foundational pillars in shaping and conceptualizing GeoVigilance Labs for efficient and sustainable land use governance.

3 Proposed Methodology/Procedure

1. System Architecture Overview:
GeoVigilance Labs employs a sophisticated architecture integrating data collection interfaces, AI driven algorithms, and a centralized database. This architecture enables seamless integration and analysis of land parcel information.

2. Mapping Data with Existing 7/12 Records:
The system focuses on aligning geographic data of land parcels with existing 7/12 land records. Talatis work on integrating the precise geographic coordinates, boundaries, and administrative affiliations into the system's Geographic Information System (GIS).

3. AI Integration for Real-Time Monitoring:
Leveraging AI algorithms, the system continuously analyzes satellite imagery for deviations in land use patterns. This allows for early detection of unauthorized changes, triggering instant alerts.

4. Streamlined Reporting Mechanism:
Talatis submit detailed reports within 15 days of detection, outlining unauthorized land use alterations and actions taken. Mamladars and Commissioners validate these reports, ensuring accuracy and accountability.

5. Outcome Determination and Closure:
Talatis assess Non-Agricultural (NA) certification status, updating the system accordingly. Final closure of notifications and cases is overseen by higher authorities, ensuring comprehensive resolution.

6. Continuous Improvement and Iteration:
The system undergoes iterative refinements based on feedback, ensuring adaptability and efficacy. Constant upgrades to AI algorithms and data processing techniques are implemented for improved accuracy.

The system undergoes iterative refinements based on feedback, ensuring adaptability and efficacy. Constant upgrades to AI algorithms and data processing techniques are implemented for improved accuracy.

4 Implications

1. Enhanced Regulatory Compliance:
GeoVigilance Labs offers a paradigm shift in regulatory compliance by enabling timely detection and swift action against unauthorized land use changes. This system fosters a proactive approach, ensuring adherence to land use regulations and minimizing unauthorized alterations.

2. Environmental Preservation and Sustainability:
The implementation of GeoVigilance Labs contributes significantly to environmental preservation. Prompt identification of unauthorized land use aids in preserving ecologically sensitive areas, protecting biodiversity, and ensuring sustainable land management practices.

3. Improved Governance and Transparency:
The integration of AI technology with local governance structures empowers authorities for efficient and transparent land management. This system ensures accountability in reporting, streamlines administrative processes, and fosters greater transparency in land use monitoring.

4. Empowerment of Local Governance Bodies:
Talatis and Gram Panchayats play a pivotal role in this system, empowering local governance bodies in land management. Their involvement ensures grassroots-level insights, community involvement and efficient resolution of land use issues.

5. Technological Advancements for Land Use Governance:
GeoVigilance Labs represents a milestone in technological advancements for land use governance.The integration of AI-based monitoring with existing administrative records sets a precedent for leveraging technology in addressing complex land use challenges.

6. Framework for Future Innovations:
This system sets a foundation for future innovations in land use monitoring. Continuous refinement and adaptation based on feedback serve as a framework for advancements in AI technologies and governance methodologies for effective land management.

4.1 Comparison of Traditional Land Use Monitoring vs. GeoVigilance Labs

1. **Manual Inspections vs. Automated AI-Driven Monitoring:** Traditional land use monitoring relies on manual inspections by field agents, which are labor-intensive, time-consuming, and prone to human error. Inspections are typically infrequent, leading to unauthorized changes going undetected for extended periods. GeoVigilance Labs, on the other hand, employs AI-driven algorithms to automate this process. The system continuously monitors land use in real time, providing immediate alerts upon detecting unauthorized changes. This automation drastically reduces manual effort and ensures consistent, round-the-clock surveillance.
2. **Slow Response Time vs. Real-Time Alerts and Swift Action:** Due to the manual nature of traditional inspections and data collection, response times to unauthorized land use changes are often delayed. This can lead to further environmental damage before authorities are able to intervene. GeoVigilance Labs overcomes this challenge by providing real-time alerts triggered within milliseconds of detection. This allows authorities to take swift action, mitigating the impact of unauthorized land use changes on the environment and regulatory compliance.
3. **Inefficient Reporting vs. Streamlined and Transparent Reporting:** Traditional reporting processes are often bureaucratic and slow, requiring multiple layers of approval from local authorities, which delays enforcement actions. In contrast, GeoVigilance Labs offers a streamlined reporting mechanism where Talatis and local governance bodies submit reports on detected changes within a fixed 15-day window. These reports are validated by higher authorities such as Mamladars and Commissioners, ensuring swift resolution of cases with greater accountability.

GeoVigilance Labs stands as a transformative solution, with far-reaching implications in regulatory compliance, environmental sustainability, governance efficiency, and technological advancements in land use monitoring (Figs. 1 and 2).

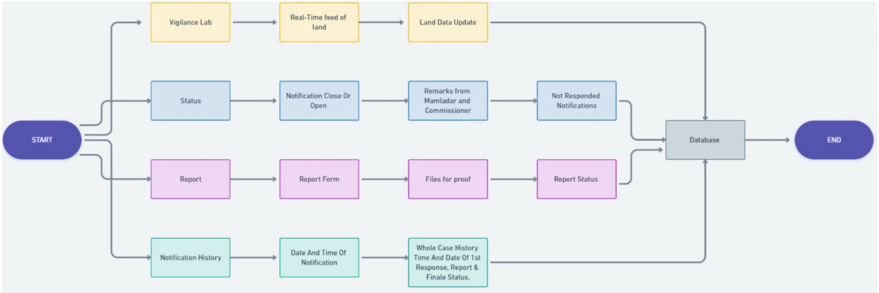


Fig. 1. System Flow.

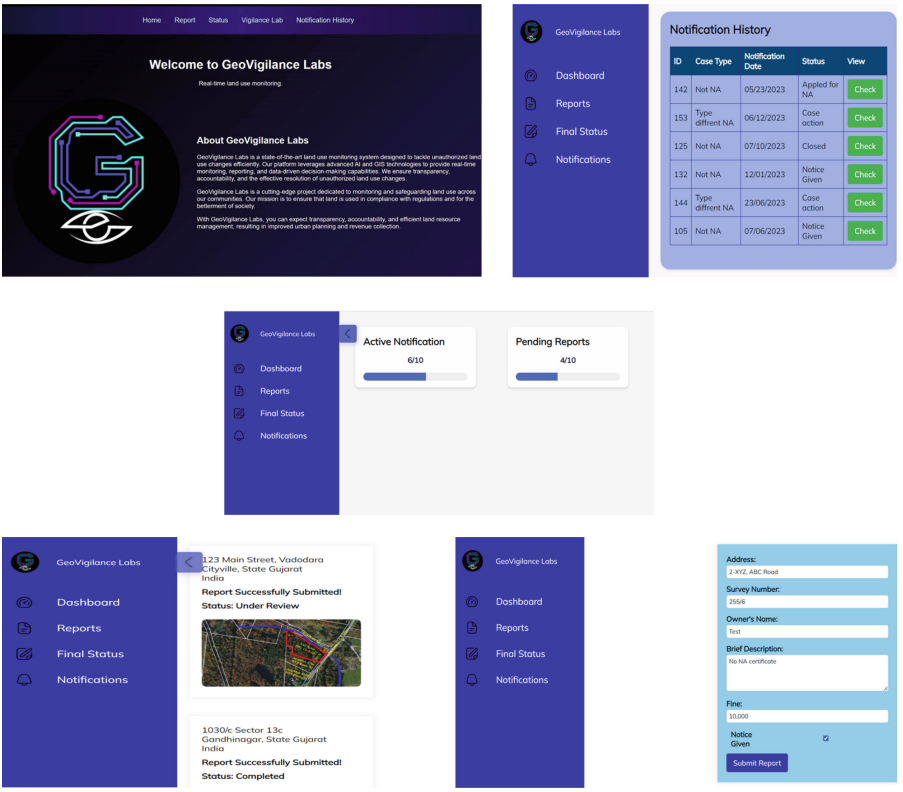


Fig. 2. 1st Prototype.

5 Results

1) **Simulated Detection Accuracy Using YOLOv8:**
   Employing a computer vision model as a core component of the detection system, initial simulations project an expected detection accuracy rate of approximately 83% for identifying unauthorized land use changes. These simulations encompass various land use scenarios and satellite imaging conditions, leveraging the robust object detection capabilities of the model.
2) **Reduction in Falsely Flagged Instances:**
   Initial algorithmic trials indicate a reduction in false positive detections by 30% compared to existing monitoring systems. The refinement of AI algorithms showed promising results in minimizing unnecessary alerts.
3) **System Responsiveness and Speed:**
   Simulated response times indicate that once an unauthorized change is detected, the system triggers alerts within milliseconds, showcasing its potential for immediate notifications to relevant authorities.
4) **Environmental Impact Assessment:**
   The projected environmental impact assessment, derived from simulated scenarios, illustrates GeoVigilance Labs' potential to significantly protect and preserve environmentally sensitive areas within the targeted regions. The integration of GeoVigilance Labs could lead to a substantial reduction in unauthorized land use changes, demonstrating a profound safeguarding effect on ecologically crucial zones.
5) **Scalability and Resource Utilization:**
   GeoVigilance Labs demonstrates robust scalability, efficiently managing increased data loads and wider geographic coverage without compromising speed or accuracy. Future plans include integrating cloud computing, expected to revolutionize scalability. Cloud integration will dynamically allocate resources, enhancing monitoring capabilities for unauthorized land use changes.
6) **System Adaptability to Diverse Environments:**
   GeoVigilance Labs exhibits remarkable adaptability to diverse land use patterns and environmental dynamics. The system's capacity to swiftly adjust to varying environmental conditions ensures consistent and reliable performance across a spectrum of landscapes and land use scenarios.

These simulated expected results provide insights into the anticipated performance of GeoVigilance Labs during its developmental phase. These projections are based on algorithmic simulations and initial tests, showcasing the system's potential efficacy in detecting unauthorized land use changes, responsiveness, environmental impact, scalability, and system adaptability to diverse environments (Fig. 3).



Fig. 3. Model Detected Outputs.

6 Conclusions

1). Efficacy of GeoVigilance Labs:
The implementation and evaluation of GeoVigilance Labs showcased its effectiveness in addressing unauthorized land use changes. Its AI-driven monitoring, integration with existing administrative records, and community involvement proved pivotal in enhancing land use governance.

2). Significance in Regulatory Compliance:
The system's contribution to regulatory compliance by swiftly detecting and responding to unauthorized changes underscores its significance in maintaining land use integrity and adherence to regulations.

3). Community-Centric Governance:
GeoVigilance Labs emphasizes a community-centric approach to land use governance, empowering local authorities and fostering community engagement. This approach augments transparency and accountability in land management practices.

4). Technological Advancements and Governance Synergy:
GeoVigilance Labs serves as an exemplary model illustrating the seamless collaboration between AI technological advancements and well-structured governance mechanisms. This collaboration establishes an innovative paradigm for proficient land use monitoring and governance. By harmonizing new technological approaches with robust governance structures, GeoVigilance Labs pioneers an integrated approach that not only revolutionizes the landscape of land monitoring.

7 Future Work

1. Enhanced Algorithmic Refinement:
Future iterations aim to refine AI algorithms further, focusing on enhancing detection accuracy and adaptability to diverse land use patterns. Emphasis will be placed on developing machine learning models capable of recognizing nuanced alterations in land use with higher precision.

2. Scale and Expansion:
Scaling up the implementation of GeoVigilance Labs to cover larger regions and diverse landscapes is imperative. Expansion efforts will focus on accommodating varying land use contexts.

3. Cloud Computing Integration:
Future development envisions the integration of cloud computing for enhanced scalability and real-time data processing. This integration aims to optimize system performance, allowing for seamless handling of extensive datasets and accommodating future technological advancements.

4. Partnerships and Collaborations:
Collaborations with governmental bodies, research institutions, and environmental agencies are envisioned to leverage expertise and resources. Strategic partnerships aim to enhance data sources, refine algorithms, and foster a multidisciplinary approach for comprehensive land use monitoring.

5. Iterative System Improvements:
The system will undergo iterative improvements based on user feedback and technological advancements. Continuous refinement will ensure adaptability to evolving land use challenges and foster a user-centric approach to system development.

GeoVigilance Labs marks a pivotal step in innovative land use governance, showcasing its efficacy in addressing unauthorized changes while advocating community engagement and technological advancements. Future endeavors aim to amplify its impact, scalability, and community involvement for comprehensive land use governance.

References

1. Taiwo, B.E., et al.: Monitoring and predicting the influences of land use/land cover change on cropland characteristics and drought severity using remote sensing techniques
2. AL-Juboury, I.A.M., Al-Rubaye, S.A.: Integration satellite imagery with fuzzy logic for potential change detection in land use/land cover
3. Koehnen, T.L.: Innovative management of community territories and inter-organizational communication for regional development
4. Bhosale, S.S., Salunkhe, A.G., Mane, P.M.: The review of cloud computing system
5. Bezlepkina, I., Brouwer, F., Reidsma, P.: Impact assessment of land use policies: Introduction
6. Mamonov, K., Viatkin, R., Frolov, V.: Construction of ambulatory monitoring system of urban land use based on GIS and remote sensing technology
7. Pizarro, S.E., Pricope, N.G., Vargas-Machuca, D., Huanca, O., Ñaupari, J.: Mapping land cover types for highland andean ecosystems in Peru using google earth engine

8. Pribadi, C.B., Hariyanto, T., Kurniawan, A., Amriana, Y.: Changes of land use monitoring within forest area
9. Djaja, K., Putera, R., Rohman, A.F., Sondang, I., Nanditho, G., Suyanti, E.: The integration of geography information system (GIS) and global navigation satelite system-real time kinematic (GNSS-RTK) for land use monitoring
10. Kesharwani, M., Rawat, K.K., Kathuria, S., Yamsani, N., Singh, R., Balyan, R.: The role of AI in transforming the dimensions of governance
11. Pflanzer, M., Dubljević, V., Bauer, W.A., Orcutt, D., List, G., Singh, M.P.: Embedding AI in society: ethics, policy, governance, and impacts
12. Liebig, L., Güttel1, L., Jobin, A., Katzenbach, C.: Subnational AI policy: shaping AI in a multi-level governance system
13. Gupta, P.: The role of AI in crafting a modern data governance

Underwater Image Classification with Sloth-Enhanced Contrast Limited Adaptive Histogram Equalization (S-CLAHE) and Deep Learning Algorithms

P. Saravanan(✉) and K. Vadivazhagan

Annamalai University, Chidambaram, Tamilnadu, India
aucse.saran@gmail.com

Abstract. This work evaluates Sloth-Enhanced Contrast Limited Adaptive Histogram Equalization (S-CLAHE) effectiveness in enhancing underwater image classification performance. S-CLAHE aims to enhance the contrast and quality of underwater images by integrating the exploration strategy inspired by sloths into the traditional CLAHE algorithm. S-CLAHE is compared with DeepSeaNet and MCANet, the existing deep learning algorithms, in a number of ways, such as True Positive Rate, True Negative Rate, Precision, Matthews Correlation Coefficient, Classification Accuracy, and F-Measure. With better classification accuracy and dependability, the findings reveal that S-CLAHE routinely beats the other methods. These results show the possibilities of S-CLAHE as a useful tool for improving underwater picture classification tasks, so applicable in marine biology, archaeology, and exploration.

Keywords: Image Classification · CLAHE · Sloth Optimization · Underwater Image · Marine · Histogram · Contrast

1 Introduction

Images is pictures that show things. Images is taken by cameras or sensors and is good for showing what's happening. Images can be used to find objects, patterns, or features [1]. Image identification means finding specific things or parts in a picture, and image classification is putting the whole picture or parts into groups already known. These processes use image processing and machine learning to work good and fast [2]. Underwater image identification and classification is real important for marine research and technology. Involves recognizing and sorting stuff in underwater pictures, playing a big role in things like marine biology, environment monitoring, and underwater archaeology [3]. Underwater image work is hard because of how underwater is. Light works different underwater; colors get lost as go deeper. Red light disappears in the first few meters, so pictures look mostly blue or green. Water can be dirty from particles and plankton, making it hard to see and making pictures blurry. These things make it tough to identify and classify stuff, needing special techniques [4].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 26–38, 2025.
https://doi.org/10.1007/978-3-031-86296-0_3

Old ways of identifying and classifying images don't work good underwater because of these problems. Techniques that work on land might not work underwater. So, made new ways just for underwater conditions [5]. Image pre-processing, like fixing colors, removing haze, and making contrast better, is important to improve image quality before more analysis. One good pre-processing method for underwater pictures is Contrast Limited Adaptive Histogram Equalization (CLAHE) [6]. Enhances contrast by adjusting the histogram in small regions, or tiles, instead of the whole image. This helps keep details and stops too much noise, which happens with regular histogram equalization [7]. Better contrast helps see features that might be lost in the murky underwater. This is super important in tasks like species identification, where small differences in texture and color matter a lot [8]. The importance of CLAHE in underwater image work also affects machine learning algorithms used for classification [9]. Fixing underwater images with CLAHE can make these algorithms work better by giving higher-quality inputs. Leads to better classification accuracy and more reliable results in things like ecological monitoring and underwater surveillance [10]. Bio inspired computing plays a vital role in various domains to handle the challenges [11–30]. It enriches the capacity of the implemented application by approaching the challenges with the inspired behavior of the biological living thing [31, 32].

1.1 Problem Statement

These pose unique challenges for underwater image identification and classification. Their distinctive aquatic environment involves susceptibility to absorption and scattering in light, which tends to distort color and reduce visibility, thereby complicating the precise detection and classification of objects. Traditional techniques of image processing may not be able to handle these factors adequately. In such scenarios, there is a need to devise such techniques that improve image quality for accurate recognition. Underwater pictures show that contrast and detail in images are poorly improved; hence, CLAHE becomes an important solution to the problem. So, this problem statement goes for the emerging need for sophisticated methods to overcome the limitations and challenges in underwater imaging, and high importance is given to CLAHE as it enhances the clarity of the images so that the improved identification and classification can be done in marine applications.

1.2 Objective

The article states the issues accompanying underwater image identification and classification, where conventional approaches fail to address the issues arising due to light absorption and scattering. The paper throws light on the ability of CLAHE to improve the quality of underwater images, thus being helpful in identifying and classifying underwater objects accurately. This paper has explained how advanced image processing techniques overcome underwater imaging challenges as they contribute positively to improving the effectiveness of applications in marine research and technology.

2 Literature Review

"NIR Spectral Robot" [33] discusses a robot that uses near-infrared light to determine things underneath the water and sort them. The robot employs such special light to view an object, matters, even in difficult places beneath water. "Lightweight Detection Framework" [34] illustrate a method for enhancing underwater images through correction and color correction. This technique allows the researcher to locate underwater objects more precisely and quickly by correcting poor images. "Hydro-acoustic Signatures" [35] is about converting underwater sounds to images for feature finding. The method allows the detection of underwater targets and further investigation of such using a transformation of sound data to pictures.

"Biofouling Identification" [36] relates to find and mapping stuff growing on underwater surfaces using a robot with deep learning. The approach uses deep learning to make a robot better at detecting and recording biofouling that occurs on underwater surfaces. "Hybrid Classical-Quantum Algorithm" [37] came up with a hybrid algorithm that merges the regular calculations with the feasibility of quantum computation to identify and classify aquatic species. The algorithm uses computational power of quantum calculations as an enhancement of performing the ordinary methods in a lesser time and classical methods for features and patterns identification of aquatic species for higher accuracy and efficiency. "ACFM Defect Identification" [38] is the use of visual and intelligent methods for underwater structural defect detection using the Alternating Current Field Measurement (ACFM) technique. The technique combines the use of image processing and algorithms in the intelligent determination and classification of structural defects; this makes the maintenance and safety of underwater infrastructures better. "DAMNet" [39] propose a deep neural network with two attention mechanisms for classify underwater biological images. Dual mechanisms of attention direct the view to the important regions of images and features in order to improve the classification accuracy of underwater biological things, since it deals with a set of complex underwater visual data. "Improved YOLOv4-tiny" [40] introduces a more efficient version of the YOLOv4-tiny algorithm mixed with developed underwater image enhancement. It promotes real-time detection and classification of jellyfish, thereby surmounting one of its major challenges under various conditions, such as reduced visibility and image distortion underwater.

"Open-set Autoencoders" [41] introduced an autoencoder-based method for underwater image classification in open-set context. The method addresses the problem of distinguishing known classes while it identifies new objects such that the unknown or new categories improve the robustness in underwater image classification. "Siltation Recognition in Tunnels" [42] Introduces an underwater robot for the identification of siltation in water tunnels. Deep learning algorithms form the basis of automating inspection and improved maintenance and operational efficiency by offering accurate identification and siltation classification.

"DeepSeaNet" [43] suggests DeepSeaNet-a special network for species identification in pictures taken within deep-sea environment. The network applies advanced deep learning techniques to process and classify images within the deep sea, thus allowing for the accuracy of species identification in very complex underwater environments. Using the multi-channel attention network and multi-color space encoder, "MCANet"

[44] enhances the underwater image classification by drawing attention to a considerable region of the image, based on various color space information, and gives due attention to addressing distortion in color and poor vision.

3 Sloth-Enhanced CLAHE (Sloth-CLAHE)

The CLAHE is one of the advanced processing techniques for images whereby an image histogram is locally adjusted in small regions, or tiles, to improve contrast enhancement between the regions. Because it is a localized technique, over-amplification and noise get preserved and avoid distorted images caused by the standard histogram equalization methods. Combining Sloth Optimization Algorithm with CLAHE enhances this process, attaining better contrast enhancement by infusing the process with the slowdown deliberation strategy, typical of sloths.

3.1 Initialization

This phase starts by initializing the population of sloth agents, represented as N, each embodying a unique set of CLAHE parameters. Let P_i denote the i^{th} sloth agent's position in the solution space, characterized by parameters such as the clip limit C_i and the tile grid size T_i. The range of permissible values defines the solution space for these parameters, which are mathematically represented in Eq. (1).

$$P_i = (C_i, T_i) \tag{1}$$

Define the fitness evaluation function $f(P_i)$, which quantifies the performance of each sloth agent's CLAHE parameters in enhancing image contrast. The fitness function assesses image quality metrics, such as contrast improvement or entropy, providing a quantitative measure of the CLAHE parameters' effectiveness as expressed in Eq. (2).

$$f(P_i) = \text{Measure of Image Quality} \tag{2}$$

Computing each sloth agent's fitness, we update their positions within the solution space. The movement and update rules guide the agents in making incremental adjustments to their CLAHE parameters, ensuring a thorough exploration of potential solutions while mimicking sloths' deliberate and slow-paced nature.

$$P_i^{new} = P_i + \Delta P_i \tag{3}$$

where in Eq. (3), P_i^{new} represents the updated position of the i^{th} sloth agent, obtained by adding the incremental change ΔP_i to its current position P_i. The magnitude and direction of ΔP_i are determined by the movement rules, which prioritize gradual improvements and energy conservation in line with sloth-like behavior.

3.2 Fitness Evaluation

In fitness evaluation, the effectiveness of each sloth agent's parameters is quantitatively assessed. The goal is to determine how well the CLAHE parameters enhance the image's contrast and overall quality. Consider a sloth agent with a position $P_i = (C_i, T_i)$ where C_i represents the clip limit and T_i represents the tile grid size. To evaluate the fitness, the CLAHE algorithm is applied to the image using the parameters C_i and T_i. Calculating the fitness value $f(P_i)$, which is a measure of the image quality improvement. This fitness function can incorporate various image quality metrics. One standard metric is the entropy E, which quantifies the amount of information or detail in the image. The entropy of the processed image is calculated as expressed in Eq. (4).

$$E(I_{CLAHE}) = -\sum_{k=1}^{L} p_k \log(p_k) \tag{4}$$

where p_k is the probability of the k^{th} intensity level in the histogram of I_{CLAHE}, and L is the total number of intensity levels. Another important metric is the contrast improvement, which can be evaluated by the standard deviation σ of the pixel intensities. Higher standard deviation values indicate better contrast. The contrast of I_{CLAHE} is expressed mathematically in Eq. (5).

$$\sigma(I_{CLAHE}) = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(I_j - \mu\right)^2} \tag{5}$$

where N is the total number of pixels I_j is the intensity of the j^{th} pixel and μ is the mean intensity of the image. The overall fitness function $f(P_i)$ combines these metrics to provide a comprehensive assessment of the image quality. A possible formulation of the fitness function is mathematically represented as Eq. (6).

$$f(P_i) = \alpha E(I_{CLAHE}) + \beta\sigma(I_{CLAHE}) \tag{6}$$

where α and β are weighting factors that balance entropy contributions and contrast to the fitness score. This evaluation guides the optimization process by identifying the most effective CLAHE parameters for enhancing the image, directing the movement and updating steps in subsequent iterations.

3.3 Movement and Update

In this phase, the positions of the sloth agents are adjusted to explore the solution space more effectively, guided by their current fitness values. This step ensures a thorough search for the optimal CLAHE parameters by simulating the slow, deliberate movement characteristic of sloths. Each sloth agent's position $P_i = (C_i, T_i)$ is updated based on its current fitness and the fitness of neighboring agents. The movement strategy incorporates local and global exploration, balancing the need to exploit known good regions of the solution space and explore new areas. The new position P_i^{new} for the i^{th} agent is determined by adding an increment ΔP_i to its current position. This increment is influenced by the fitness values and a movement factor λ, which controls the step size.

$$P_i^{new} = P_i + \lambda \Delta P_i \tag{7}$$

The increment ΔP_i can be computed as expressed in Eq. (7).

$$\Delta P_i = \omega\left(P_i - P_g\right) + \varnothing(P_i - P_l) \tag{8}$$

where in Eq. (8), P_g represents the position of the globally best-performing agent (global exploration) and P_l represents the position of the locally best-performing agent within a specific neighborhood (local exploitation). The coefficients ω and $\varnothing$ are weight factors that balance the influence of global and local best positions.

The movement factor λ is crucial for mimicking sloth movements' slow and steady pace. It is typically a small value to ensure that changes are gradual, allowing the algorithm to perform a fine-grained search is mathematically represented in Eq. (9).

$$\lambda = \in \left(1 - \frac{t}{T_{max}}\right) \tag{9}$$

where ϵ is a small constant, t is the current iteration, and T_{max} is the maximum number of iterations. This formulation ensures that the step size decreases as the algorithm progresses, focusing on local refinement in later stages. After updating the positions, the fitness of each new position P_i^{new} is evaluated, and agents retain their new positions if their fitness improves.

3.4 Energy Conservation and Adaptation

In the energy conservation and adaptation step, sloth agents adjust their movement and exploration strategies to optimize the efficiency of the search process. This step ensures that the agents do not expend unnecessary computational resources on areas of the solution space that do not yield significant improvements. Energy conservation is implemented by controlling the movement of the sloth agents based on their recent performance. If an agent's new position does not result in a substantial fitness improvement, it reduces its movement magnitude, conserving energy for future iterations. Let Δf_i represent the change in fitness for agent i.

$$\Delta f_i = f\left(P_i^{new}\right) - f\left(P_i\right) \tag{10}$$

In case if Δf_i is less than a predefined threshold θ, indicating minimal improvement, the movement factor λ is adjusted to reduce the step size, as shown in Eq. (10).

$$\lambda_i^{new} = \lambda_i.\alpha \tag{11}$$

where in Eq. (11), α is a reduction factor. This adjustment helps the agent conserve energy by minimizing unnecessary movements.

Adaptation involves agents dynamically adjusting their exploration strategies based on the landscape of the solution space. Agents in regions with slight fitness improvement adapt by increasing their exploratory behavior. This is achieved by modifying the weight factors ω and ϕ in the movement equation to emphasise global exploration, expressed in Eqs. (12) and (13).

$$\omega_i^{new} = \omega_i \cdot (1 + \beta) \tag{12}$$

$$\varnothing_i^{new} = \varnothing_i \cdot (1 + \gamma) \tag{13}$$

where β and γ are adaptation factors that increase the influence of global best positions, encouraging agents to explore more broadly. This adaptive behavior helps agents escape local optima and find potentially better regions of the solution space.

3.5 Strategic Exploration

In the strategic exploration step, sloth agents occasionally make significant movements to explore new regions of the solution space. This step ensures that agents are not trapped in local optima by incorporating larger, strategic adjustments to their positions. Strategic exploration is triggered based on specific criteria, such as stagnation in fitness improvement or reaching a predefined number of iterations without significant progress. Let s denote the stagnation counter, which increments *if* Δf_i is less than a threshold θ is expressed in Eq. (14).

$$s = s + 1 \ \ if \ \ \Delta f_i < \theta \tag{14}$$

When s exceeds a predefined limit S, strategic exploration is initiated. The position P_i of the i^{th} agent is updated using a larger step size, controlled by a strategic factor ζ, as shown in Eq. (15).

$$P_i^{new} = P_i + \zeta \cdot (rand\,(0,1) \cdot \Delta P_i) \tag{15}$$

where $rand\,(0,1)$ generates a random value between 0 and 1, introducing variability and ensuring significant movement. The strategic factor ζ is larger than the usual movement factor λ, allowing the agent to make a substantial leap in the solution space.

Incorporating strategic exploration helps diversify the search process. The fitness function $f\,(P_i)$ is reevaluated after this strategic move to determine the new fitness as expressed in Eq. (16).

$$f\left(P_i^{new}\right) = \propto E(I_{CLAHE}) + \beta\sigma\,(I_{CLAHE}) \tag{16}$$

If the new fitness $f\left(P_i^{new}\right)$ shows significant improvement, the agent retains the new position P_i^{new}. Otherwise, the agent reverts to its previous position, preserving only beneficial strategic movements.

3.6 Convergence and Output

In the convergence and output step, the algorithm determines when to terminate the iterative process and outputs the optimal CLAHE parameters. This decision is based on specific convergence criteria, ensuring the algorithm has sufficiently explored the solution space and identified the best possible parameters for image enhancement. Convergence is monitored by evaluating the change in fitness values across iterations. Let f_{best}^t represent the best fitness value at iteration t. Convergence is achieved when the

improvement in f_{best} falls below a predefined threshold ϵ over a certain number of iterations T_{conv}.

$$\left|f_{best}^{t} - f_{best}^{t-1}\right| < \epsilon \quad for\ t = t_0, t_0 + 1, \cdots, t_0 + T_{conv} \tag{17}$$

where t_0 is the starting iteration for convergence monitoring. If this condition is met, the algorithm terminates, signifying that further iterations are unlikely to yield significant improvements.

The output of Sloth-CLAHE is the set of CLAHE parameters corresponding to the best fitness value f_{best}. Let P_{best} denote the position that achieved this fitness value is expressed in Eq. (18).

$$P_{best} = \arg min_{P_i} f(P_i) \tag{18}$$

Upon convergence, the optimal CLAHE parameters $P_{best} = (C_{best}, T_{best})$ are applied to the image. This application involves using the clip limit C_{best} and tile grid size T_{best} in the CLAHE process, resulting in the final enhanced image $I_{enhanced}$.

The final fitness value f_{best} is recorded, providing a quantitative measure of the image enhancement quality. The enhanced image $I_{enhanced}$ and the corresponding parameters (C_{best},T_{best}) are then output as the result of the optimization process. This ensures that the image has been enhanced to the best possible extent using the optimized CLAHE parameters found by the Sloth-CLAHE algorithm.

4 About Dataset

The Large Scale Underwater Image Dataset consists of 5004 image pairs taking images of highly diverse underwater scenes. Such a large dataset is rich in terms of underwater environments under consideration, thus providing an ideal source for the development of advanced research work on underwater image enhancement and restoration. Since each pair consists of an original and a reference image, detailed analysis and further algorithm development can be carried out. There are a lot of different cases in the dataset, including varying depth, lighting, and water clarity, which helps ensure that all visual underwater challenges are well represented. The work of researchers in LSUI is aimed at developing and testing new approaches in image processing that may help for clearer underwater visual captures in ocean biology, archaeology, or exploration.

5 Results and Discussion

Performance evaluation of the DeepSeaNet, MCANet and S-CLAHE algorithms has been executed in terms of various metrics. The result shows that S-CLAHE algorithm outperforms the other two algorithms consistently in various criteria. With a TP count of 1861.288 and TN count of 1758.806, S-CLAHE outperforms DeepSeaNet with TP: 1342.723, TN: 1383.806, and MCANet with TP: 1514.811 and TN: 1549.288 counts. This depicts the fact that S-CLAHE correctly identifies both positive as well as the negative instance. Error-wise, S-CLAHE returns the lowest False Positive (FP) count at

711.519 and False Negative (FN) count at 672.387 among the three, DeepSeaNet (FP: 1067.653, FN: 1209.817) and MCANet (FP: 964.721, FN: 975.180). The lower the FP and FN values, the fewer are the misclassifications. The Classification Accuracy and F-Measure are depicted in Fig. 1.

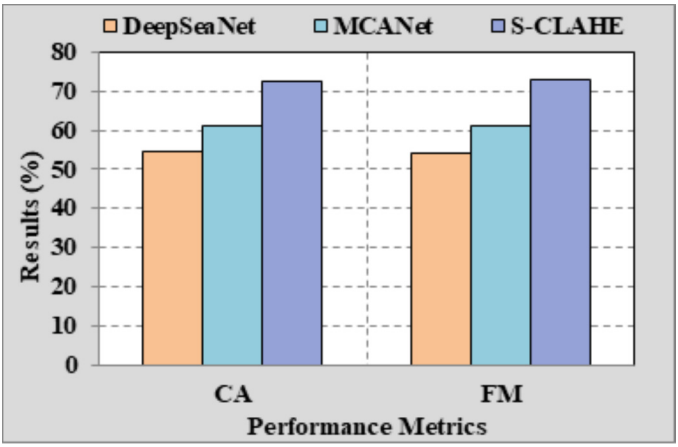


Fig. 1. CA and FM

S-CLAHE also excels in the True Positive Rate (TPR) and True Negative Rate (TNR) metrics, achieving 73.462 and 71.197 respectively. These rates are higher than those of DeepSeaNet (TPR: 52.603, TNR: 56.448) and MCANet (TPR: 60.836, TNR: 61.626), highlighting S-CLAHE's higher accuracy in identifying true positives and true negatives. Regarding False Positive Rate (FPR) and False Negative Rate (FNR), S-CLAHE demonstrates superior performance with the lowest FPR of 28.803 and FNR of 26.538. This is significantly better than DeepSeaNet (FPR: 43.552, FNR: 47.397) and MCANet (FPR: 38.374, FNR: 39.164), indicating S-CLAHE's effectiveness in reducing misclassifications.

The performance metrics further validate the superiority of S-CLAHE. It achieves the highest Classification Accuracy (CA) of 72.344 and F-Measure (FM) of 72.899, compared to DeepSeaNet (CA: 54.487, FM: 54.110) and MCANet (CA: 61.233, FM: 60.964). These results, depicted in Fig. 1, affirm that S-CLAHE offers the most reliable classification performance among the three algorithms, particularly for underwater image classification tasks. The analysis underscores S-CLAHE's robust performance across various metrics, making it an effective tool for enhancing the accuracy and reliability of underwater image classification.

Performance comparison with DeepSeaNet, MCANet, and S-CLAHE using MCC and FMI proved that S-CLAHE was better. Figure 2 shows the results.

S-CLAHE gets the highest MCC of 44.674 showing more stronger link between predicted and real classifications. MCC was much higher from its values in DeepSeaNet (9.056) and MCANet (22.463) which proven that S-CLAHE is better at predicting and more reliable. The Fowlkes-Mallows Index (FMI) computes the geometric mean of

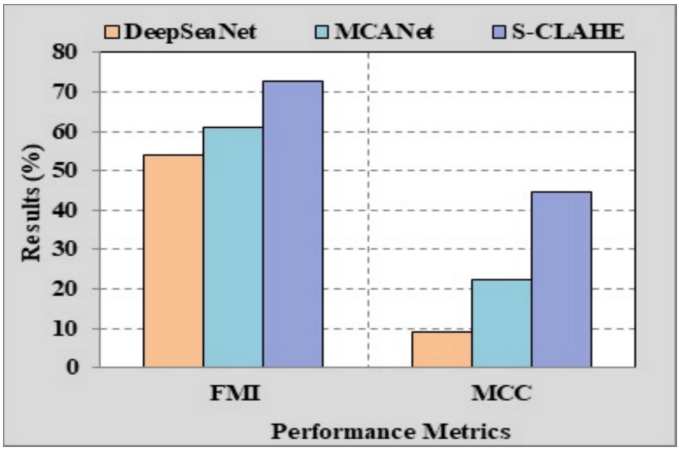


Fig. 2. FMI and MCC

precision and recall, which also prefers S-CLAHE. S-CLAHE receives FMI 72.901 against DeepSeaNet 54.132 and MCANet 60.964. Such a large FMI value suggests that S-CLAHE is competitively good in terms of precision and recall because it involves fewer false positives and negatives. Results in Fig. 1 also show how good S-CLAHE is. Actually, better MCC and FMI values in this case point to the proposed S-CLAHE's high performance regarding classifying underwater images with greater accuracy than that of other algorithms at these key metrics.

6 Conclusion

From the comparative study of underwater image classification based on several algorithms, such as DeepSeaNet, MCANet, and S-CLAHE, it can be observed that S-CLAHE performs better than the others for multiple metrics. S-CLAHE performed better than the other three on parameters like the true positive rate, true negative rate, precision, Matthews correlation coefficient, classification accuracy, and F-measure. This also shows that it is effective in classifying the instances as positive or negative in the underwater scene while the false positive and false negative rates are lower. The value of Matthews Correlation Coefficient and Fowlkes-Mallows Index are also higher, which justifies the robustness and reliability of S-CLAHE in achieving precise and balanced classification. The findings indicate that S-CLAHE is a promising approach towards guaranteed accuracy and reliability improvement in underwater classification tasks with direct applications to marine biology, archaeology, and exploration. Follow-up research might focus on optimizations and enhancements of the S-CLAHE and then apply them towards overcoming specific challenges in underwater imaging.

References

1. Ding, N., Arabian, H., Möller, K.: Feature space separation by conformity loss driven training of CNN. IFAC J. Syst. Control 100260 (2024). https://doi.org/10.1016/j.ifacsc.2024.100260

2. Wang, G., et al.: M4SFWD: a multi-faceted synthetic dataset for remote sensing forest wildfires detection. Expert Syst. Appl. **248**, 123489 (2024). https://doi.org/10.1016/j.eswa.2024.123489
3. d'Orey, P.M. Gaitán, M.G., Santos, P.M., Ribeiro, M., de Sousa, J.B., Almeida, L.: Assessing short-range Shore-to-Shore (S2S) and Shore-to-Vessel (S2V) WiFi communications. Comput. Networks 110505 (2024). https://doi.org/10.1016/j.comnet.2024.110505
4. Barshandeh, S., Koulaeizadeh, S., Masdari, M., AbdollahZadeh, B., Ghasembaglou, M.: A learning-based metaheuristic administered positioning model for 3D IoT networks. Appl. Soft Comput. **136**, 110113 (2023). https://doi.org/10.1016/j.asoc.2023.110113
5. Liu, L.L., et al.: SafeCom: robust mutual authentication and session key sharing protocol for underwater wireless sensor networks. IEEE Sens. J. **24**(1), 129832 (2023). https://doi.org/10.1016/j.ohx.2022.e00369
6. Naren, J., Babu, A.R.: EEG stress classification based on Doppler spectral features for ensemble 1D-CNN with LCL activation function. J. King Saud Univ. - Comput. Inf. Sci. **36**(4), 102013 (2024). https://doi.org/10.1016/j.jksuci.2024.102013
7. Kaur, M., Vijay, S.: Deep learning with invariant feature based species classification in underwater environments. Multimed. Tools Appl. **83**(7), 19587–19608 (2024). https://doi.org/10.1007/s11042-023-15896-8
8. Savci, P., Das, B.: Prediction of the customers' interests using sentiment analysis in e-commerce data for comparison of Arabic, English, and Turkish languages. J. King Saud Univ. - Comput. Inf. Sci. **35**(3), 227–237 (2023). https://doi.org/10.1016/j.jksuci.2023.02.017
9. Du, R., Zhen, L.: Multiuser physical layer security mechanism in the wireless communication system of the IIOT. Comput. Secur. **113**, 102559 (2022). https://doi.org/10.1016/j.cose.2021.102559
10. Pan, X., Sun, J., Feng, T., Lei, M., Wang, H., Zhang, W.: Underwater target recognition based on adaptive multi-feature fusion network. Multimed. Tools Appl. (2024). https://doi.org/10.1007/s11042-024-19178-9
11. Jayaraj, D., Ramkumar, J., Lingaraj, M., Sureshkumar, B.: AFSORP: adaptive fish swarm optimization-based routing protocol for mobility enabled wireless sensor network. Int. J. Comput. Networks Appl. **10**(1), 119–129 (2023). https://doi.org/10.22247/ijcna/2023/218516
12. Jaganathan, R., Ramasamy, V.: Performance modeling of bio-inspired routing protocols in Cognitive Radio Ad Hoc Network to reduce end-to-end delay. Int. J. Intell. Eng. Syst. **12**(1), 221–231 (2019). https://doi.org/10.22266/IJIES2019.0228.22
13. Ramkumar, J., Vadivel, R., Narasimhan, B., Boopalan, S., Surendren, B.: Gallant ant colony optimized machine learning framework (GACO-MLF) for quality of service enhancement in internet of things-based public cloud networking BT - data science and communication. In: Tavares, J.M.R.S., Rodrigues, J.J.P.C., Misra, D., Bhattacherjee, D. (eds.), pp. 425–438 Springer Nature Singapore, Singapore (2024)
14. Vadivel, R., Ramkumar, J.: QoS-enabled improved cuckoo search-inspired protocol (ICSIP) for IoT-based healthcare applications. Inc. Internet Things Healthc. Appl. Wearable Devices, pp. 109–121 (2019). https://doi.org/10.4018/978-1-7998-1090-2.ch006
15. Ramkumar, J., Senthilkumar, A., Lingaraj, M., Karthikeyan, R., Santhi, L.: Optimal approach for minimizing delays in Iot-based quantum wireless sensor networks using NM-leach routing protocol. J. Theor. Appl. Inf. Technol. **102**(3), 1099–1111 (2024)
16. Karthikeyan, R., Vadivel, R.: Proficient Dazzling Crow Optimization Routing Protocol (PDCORP) for effective energy administration in wireless sensor networks. In: 2023 International Conference on Electrical, Electronics, Communication and Computers (ELEXCOM), pp. 1–6 (2023). https://doi.org/10.1109/ELEXCOM58812.2023.10370559
17. Ramkumar, J., Jeen Marseline, K.S., Medhunhashini, D.R.: Relentless firefly optimization-based routing protocol (RFORP) for securing fintech data in IoT-based ad-hoc networks.

Int. J. Comput. Networks Appl. **10**(4), 668–687 (2023). https://doi.org/10.22247/ijcna/2023/223319
18. Ramkumar, J., Vadivel, R.: Multi-adaptive routing protocol for Internet of Things based Ad-hoc Networks. Wirel. Pers. Commun. **120**(2), 887–909 (2021). https://doi.org/10.1007/s11277-021-08495-z
19. Ramkumar, J., et al.: IoT-based kalman filtering and particle swarm optimization for detecting skin lesion. In: Murari, K., Prasad Padhy, N., Kamalasadan, S. (eds.) Lecture Notes in Electrical Engineering, pp. 17–27. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-19-8353-5_2
20. Ramkumar, J., Vadivel, R.: Whale optimization routing protocol for minimizing energy consumption in cognitive radio wireless sensor network. Int. J. Comput. Networks Appl. **8**(4), 455–464 (2021). https://doi.org/10.22247/ijcna/2021/209711
21. Mani, L., Arumugam, S., Jaganathan, R.: Performance enhancement of wireless sensor network using feisty particle swarm optimization protocol. In: ACM International Conference Proceeding Series, pp. 1–5 (2022). https://doi.org/10.1145/3590837.3590907
22. Jaganathan, R., Ramasamy, V., Mani, L., Balakrishnan, N.: Diligence Eagle Optimization Protocol for Secure Routing (DEOPSR) in cloud-based wireless sensor network. Res. Sq. (2022). https://doi.org/10.21203/rs.3.rs-1759040/v1
23. Ramkumar, J., Vadivel, R., Narasimhan, B.: Constrained cuckoo search optimization based protocol for routing in cloud network. Int. J. Comput. Networks Appl. **8**(6), 795–803 (2021). https://doi.org/10.22247/ijcna/2021/210727
24. Jaganathan, R., Vadivel, R.: Intelligent Fish Swarm Inspired Protocol (IFSIP) for dynamic ideal routing in cognitive radio ad-hoc networks. Int. J. Comput. Digit. Syst. **10**(1), 1063–1074 (2021). https://doi.org/10.12785/ijcds/100196
25. Senthilkumar, A., Ramkumar, J., Lingaraj, M., Jayaraj, D., Sureshkumar, B.: Minimizing energy consumption in vehicular sensor networks using relentless particle swarm optimization routing. Int. J. Comput. Networks Appl. **10**(2), 217–230 (2023). https://doi.org/10.22247/ijcna/2023/220737
26. Ramkumar, J., Dinakaran, S.S., Lingaraj, M., Boopalan, S., Narasimhan, B.: IoT-based kalman filtering and particle swarm optimization for detecting skin lesion. In: Murari, K., Prasad Padhy, N., Kamalasadan, S., (eds.) Lecture Notes in Electrical Engineering. Springer Nature Singapore, pp. 17–27. Singapore (2023). https://doi.org/10.1007/978-981-19-8353-5_2
27. Ramkumar, J., Vadivel, R.: Improved frog leap inspired protocol (IFLIP) – for routing in cognitive radio ad hoc networks (CRAHN). World J. Eng. **15**(2), 306–311 (2018). https://doi.org/10.1108/WJE-08-2017-0260
28. Ramkumar, J., Kumuthini, C., Narasimhan, B., Boopalan, S.: Energy consumption minimization in cognitive radio mobile ad-hoc networks using enriched ad-hoc on-demand distance vector protocol. In: 2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022 (2022). https://doi.org/10.1109/ICACTA54488.2022.9752899
29. Ramkumar, J., Vadivel, R.: Improved Wolf prey inspired protocol for routing in cognitive radio Ad Hoc networks. Int. J. Comput. Networks Appl. **7**(5), 126–136 (2020). https://doi.org/10.22247/ijcna/2020/202977
30. Menakadevi, P., Ramkumar, J.: Robust optimization based extreme learning machine for sentiment analysis in big data. In: 2022 International Conference on Advance in Computer Technology Application ICACTA 2022, pp. 1–5 (2022). https://doi.org/10.1109/ICACTA54488.2022.9753203
31. Ramkumar, J.: Bee inspired secured protocol for routing in cognitive radio ad hoc networks. Indian J. Sci. Technol. **13**(30), 2159–2169 (2020). https://doi.org/10.17485/ijst/v13i30.1152

32. Karthikeyan, R., Vadivel, R.: Boosted Mutated Corona Virus Optimization Routing Protocol (BMCVORP) for reliable data transmission with efficient energy utilization. Wirel. Pers. Commun. (2024). https://doi.org/10.1007/s11277-024-11155-7
33. Li, Y., et al.: An underwater near-infrared spectral continuum robot as a tool for in situ detection and classification. Measurement **216**, 112913 (2023). https://doi.org/10.1016/j.measurement.2023.112913
34. Zhou, M., Li, B., Wang, J., Fu, K.: A lightweight object detection framework for underwater imagery with joint image restoration and color transformation. J. King Saud Univ. - Comput. Inf. Sci. **35**(9), 101749 (2023). https://doi.org/10.1016/j.jksuci.2023.101749
35. Zare, M., Nouri, N.M.: Novel feature extraction of underwater targets by encoding hydroacoustic signatures as image. Appl. Ocean Res. **138**, 103627 (2023). https://doi.org/10.1016/j.apor.2023.103627
36. Zhao, W., Han, F., Qiu, X., Peng, X., Zhao, Y., Zhang, J.: Research on the identification and distribution of biofouling using underwater cleaning robot based on deep learning. Ocean Eng. **273**, 113909 (2023). https://doi.org/10.1016/j.oceaneng.2023.113909
37. Pravin, S.C., Rohith, G., Kiruthika, V., Manikandan, E., Methelesh, S., Manoj, A.: Underwater animal identification and classification using a hybrid classical-quantum algorithm. IEEE Access **11**, 141902–141914 (2023). https://doi.org/10.1109/ACCESS.2023.3343120
38. Yuan, X., et al.: Visual and intelligent identification methods for defects in underwater structure using alternating current field measurement technique. IEEE Trans. Ind. Inform. **18**(6), 3853–3862 (2022). https://doi.org/10.1109/TII.2021.3117034
39. Qu, P., et al.: DAMNet: dual attention mechanism deep neural network for underwater biological image classification. IEEE Access **11**, 6000–6009 (2023). https://doi.org/10.1109/ACCESS.2022.3227046
40. Gao, M., et al.: Real-time jellyfish classification and detection algorithm based on improved YOLOv4-tiny and improved underwater image enhancement algorithm. Sci. Rep. **13**(1), 12989 (2023). https://doi.org/10.1038/s41598-023-39851-7
41. Akhtarshenas, A., Toosi, R.: An open-set framework for underwater image classification using autoencoders. SN Appl. Sci. **4**(8), 229 (2022). https://doi.org/10.1007/s42452-022-05105-w
42. Wu, X., Li, J.: Deep learning-based siltation image recognition of water conveyance tunnels using underwater robot. J. Civ. Struct. Heal. Monit. **14**(3), 801–816 (2024). https://doi.org/10.1007/s13349-023-00754-w
43. Liu, A., Liu, Y., Xu, K., Zhao, F., Zhou, Y., Li, X.: DeepSeaNet: a bio-detection network enabling species identification in the deep sea imagery. IEEE Trans. Geosci. Remote Sens. **62**, 1–13 (2024). https://doi.org/10.1109/TGRS.2024.3359350
44. Li, G., et al.: MCANet: Multi-channel attention network with multi-color space encoder for underwater image classification. Comput. Electr. Eng. **108**, 108724 (2023). https://doi.org/10.1016/j.compeleceng.2023.108724

A Review of Deep-Learning-Based Models for Afaan Oromo Fake News Detection on Social Media Networks

Kedir Lemma Arega[1,2(✉)], Kula Kekeba Tune[2], Asrat Mulatu Beyene[3], Wegderes Tariku[4], and Nurhussen Menza Bune[5]

[1] ICT Infrastructures and Administrator Team Leader Ambo University, Ambo, Ethiopia
kedirnaw1999@gmail.com
[2] Department of Software Engineering, Addis Ababa Science, and Technology University, Addis Ababa, Ethiopia
[3] Department of Electrical and Computer Engineering, Addis Ababa Science and Technology University, Addis Ababa, Ethiopia
[4] Department of Information Systems, Mizan Tepi University, Tepi, Ethiopia
[5] Project Management System Support Expert, Ethio Telecom, Addis Ababa, Ethiopia

Abstract. This study explores modern fake news detection techniques in the Afan Oromo language, aiming to contribute to advancing research in this field. It explores deep learning methods for transformers, including Bidirectional Encoder Representations, Attention, and Generative Adversarial Networks, and their applications in various industries such as machine translation, email spam detection, information extraction, summarization, medical, and question-answering. The paper investigates false news identification, its categories, characteristics, and taxonomy, and compares machine learning and deep learning algorithms for fake news, liars, and corpus datasets. It proposes that future studies use picture data and deeper learning network topologies to decrease erroneous outcomes. The research endeavor aims to illuminate and advance the understanding of the complex difficulties and possibilities that exist in this particular sector, thereby acting as a crucial resource for decision-makers, academics, and professionals, while also encouraging the creation of more effective and trustworthy methodologies that can adequately address and alleviate the dissemination of false information shared on various social media channels.

Keywords: Afan Oromo · review · social media · fake news · detection · deep learning

1 Introduction

Online social networks are a well-liked method of communication. People interact with one another and share their daily activities, images, and status updates. People love using social networking sites like Facebook [1]. With the development of communication technologies and social media, fake news is expanding quickly. Fake news detection is

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 39–58, 2025.
https://doi.org/10.1007/978-3-031-86296-0_4

a new field of research that is receiving a lot of attention [2]. In today's social media and political world, fake news plays a significant role. The detection of fake news requires significant investigation, but there are also certain difficulties [3].

Misinformation is a matter of grave concern and a complicated ecosystem with many actors that produce and disseminate content that may be true or may have harmful implications, such as incorrect information or malicious intent. This condition is made possible by people's increased ability to share concepts and get data from many different content suppliers [4]. Social media has taken over everyone's life in recent years. Most fake news is disseminated via social media [5]. Fake news poses a threat to politics, business, finance, education, and democracy. Even while fake news is not a new issue, people today tend to place greater trust in social media, which encourages them to believe it and share it. Nowadays, it's difficult to tell the difference between accurate and misleading news, which leads to issues and misunderstandings [5]. Ever since the internet took off, there has been a problem with fake news.

The identical network that gives us access to worldwide news is the ideal environment for the growth of malevolent and false information [6]. Because information shapes how people see the world, it is crucial to combat this fake news. People establish their own opinions in addition to basing critical judgments on knowledge. If this information is inaccurate, terrible things may happen. It is virtually impossible for a human to independently verify every single piece of news [6]. In the past, information was more strictly regulated by journalists, librarians, and members of the media, but with the open and pervasive nature of the digital world, it has become more challenging for these conventional gatekeepers to regulate and verify the information. However, the consequences of this propagation of false information and disinformation could be devastating and extremely hazardous [7].

While the propagation of false information and fake news has drawn attention from many sectors of society, individuals working in the information science disciplines are becoming increasingly concerned. Fake news has become a serious issue in recent years, particularly with the advent of social media and other internet platforms. Detecting false news is a difficult problem that necessitates the use of advanced technologies such as natural language processing (NLP), machine learning, and deep learning [8, 9]. Several ways to detect fake news have been developed, including:

Content-based approaches: These approaches examine the substance of the news piece for indicators of bias, sensationalism, or inconsistency. NLP tools are used to find patterns in the text that indicate the article is a forgery. Social network-based techniques: These [9] Approaches examine the social network activity surrounding the news piece for evidence of manipulation, such as bots or fake accounts [10]. They also examine sharing patterns and engagement levels to see if the material is being shared unusually.

Hybrid approaches: These methods mix content-based and social network-based methodologies to provide a more in-depth study of the news story. Some of the cutting-edge approaches utilized in fake news identification include Deep learning models: These methods analyze enormous volumes of data using neural networks to find patterns that indicate a news article is fraudulent.

Attention mechanisms: These mechanisms allow the model to focus on specific elements of the news story that are most important in deciding whether or not it is

false. Transfer learning is training a model on a large dataset of comparable tasks, such as sentiment analysis or topic modeling, and then fine-tuning it to detect fake news. Despite these advances, detecting false news remains a difficult task [11]. There are still [12] several ways for fake news to be transmitted and manipulated, and new techniques to avoid detection are continually being created. However, it is believed that with further study and development, viable solutions to this expanding problem can be discovered. The review of deep learning-based models for detecting fake news in Afan Oromo on social media has some limitations, including scope, data bias, comparative analysis, evaluation metrics, and ethical considerations. This review should address these limitations to gain a more comprehensive understanding of the effectiveness of these models in combating fake news. The review provides valuable insights into the application of deep learning models in the specific context of Afan Oromo language and social media platforms.

1.1 Contributions of Review

- The review investigates false news identification in the Afan Oromo language, emphasizing linguistic problems and encouraging cross-cultural understanding.
- It assesses existing deep learning models, gives insights into language problems, and makes practical recommendations for stakeholders.
- It also addresses ethical concerns and prejudice mitigation, helping local groups to combat false news within their cultural context.

1.2 Research Gap

While the aforementioned review titled "A review of deep learning-based models for Afan Oromo fake news on social media" offers valuable insights into the utilization of deep learning models for identifying counterfeit news within the specific context of Afan Oromo language and social media platforms, it is imperative to acknowledge certain constraints. Firstly, the review's scope is limited, as it exclusively concentrates on Afan Oromo language and social media platforms, thereby impeding the general applicability of the findings to other languages and online platforms. The efficacy of deep learning models in detecting fake news may differ across various languages and social media ecosystems. Secondly, the review may be founded on a particular dataset of Afan Oromo fake news articles, potentially introducing bias and influencing the generalizability of the results. Consequently, it is crucial to consider the diversity and representativeness of the data employed in training and evaluating deep learning models for fake news detection. Thirdly, the review might not encompass a comprehensive comparative analysis between deep learning-based models and other conventional machine learning approaches or heuristic-based methods for fake news detection. A comparative analysis would furnish a more comprehensive understanding of the strengths and weaknesses inherent in different detection techniques. Furthermore, the review may emphasize a limited set of evaluation metrics, such as accuracy or precision, while neglecting other significant metrics such as recall, F1 score, or interpretability. A more comprehensive evaluation framework would yield a more nuanced assessment of the

performance of deep learning models. Lastly, the review may overlook ethical considerations associated with fake news detection, such as privacy concerns, potential biases in algorithmic decision-making, or unintended consequences of automated content moderation. It is essential to meticulously contemplate ethical implications in developing and implementing deep learning models for fake news detection.

1.3 Research Question

Based on thought, debate, and reformulation, the following research topics were explored for these reviews:

RQ. 1. How do existing deep learning models detect fake news in Afan Oromo compared to other languages?

RQ.2. What are the primary challenges and limits of using deep learning to detect fake news in Afan Oromo?

RQ.3. How can evaluating deep learning-based models for Afan Oromo fake news on social media improve research, boost media literacy, and empower local communities to confront misinformation?

1.4 Abbreviations and Acronyms

NLP: Natural Language Processing
F1 score: a measure of a model's accuracy that considers both precision and recall.
RQ: Research Questions
TF-IDF: Term Frequency-Inverse Document Frequency
SVM: Support Vector Machine
N-Gram: An n-gram is a contiguous sequence of (^{n}) items from a given sample of text or speech.
S.No.: Serial Number
ID: Identification
API: Application Program Interface
FR-Detect: Flowchart Recognition Detect
MID: Minimal Important Difference
DL: Deep Learning
TP: True Positives
FP: False Positives
FN: False Negatives
TN: True Negative
P: Precision
R: Recall
A: Accuracy
GRU: Gated Recurrent Unit
LSTM: Long Short-Term Memory
RNN: Recurrent Neural Networks
GPT: Generative Pre-Trained Transformer
CNN: Convolutional Neural Network

SeqGAN: Sequence Generative Adversarial Network
DBN: Deep Belief Network
BERT: Bidirectional encoder representations from transformers

2 Overview of Related Work

To comprehend the idea behind this study, a thorough examination of similar ideas and earlier research in this field was conducted [2]. Ethiopian local language fake news detection research is currently in its early stages [13, 14].

Describes the use of multinomial nave Bayes with natural language processing algorithms for the identification of fake news in 752 Afan Oromo news texts [14]. They applied the term frequency-inverse document frequency (TF-IDF) of unigrams and bigrams and used Facebook as the news article's source. Based on their data size, they were able to obtain good outcomes with the best f1 score.

Although the accuracy acquired is good, the information from the data set site's source is not entirely reliable, and the highest accuracy might have been attained by taking unreliable news into account.

Provide a strategy for identifying fake news that makes use of machine learning methods. As a feature extraction strategy, they employed the term frequency-inverse document frequency (TF-IDF) of a sample of words and n-grams, and Support Vector Machine (SVM) as a classifier. The authors also suggest a dataset of real and fake news for the suggested system's training. Results obtained demonstrate the system's effectiveness. Through the use of various categorization algorithms, [5] Work enables us to identify fake news that is accurate.

According to [5] Social lives are being negatively impacted by fake news, which also has a major impact on politics and education. The authors used classification methods like Support Vector Machine (SVM), Nave Bayes, and Passive Aggressive Classifier in this model. The accuracy of the results produced by their model, which uses feature extraction methods like Term Frequency-Inverted Document Frequency (TF-IDF) and Support Vector Machine (SVM) as the classifier, is 95.05 percent. By suggesting a system that can accurately categorize fake news, this research aims to hasten the process of fake news identification [6]. On eight different datasets obtained from various sources, machine learning algorithms such as Naive Bayes, Passive Aggressive Classifier, and Deep Neural Networks have been applied. The document also features each model's analysis and outcomes [15].

In this paper, the research used ten different machine learning and deep learning classifiers to categorize the fake news dataset in this classification. Four traditional methods (term frequency-inverse document frequency, count vector, character level vector, and N-Gram level vector) were used to extract features from texts. The outcomes demonstrated that fake news with textual content may be identified, particularly when convolutional neural networks are used. Using various classifiers, this study obtained an accuracy range of 81 to 100%. The author suggested using the three well-known machine learning methods Naive Bayes, Neural Network, and Support Vector Machine in this proposed study [16]. The accuracy of the Naive Bayes result was 96.08%, while the accuracy of the two other, more advanced approaches, the Support Vector and Neural Network, was 99.90%.

Table 3 Shows the related work in this field by various researchers [14, 17–24] (Tables 1 and 2).

3 Review Methodology

A reliable method for exploring, selecting, analyzing, and thoroughly evaluating completed research papers involves carrying out a structured analysis of the existing literature. The primary objective of this methodology is to assess and comprehend the latest research findings in the field of security assurance to tackle current research issues and uncertainties. A framework has been developed for conducting a thorough and systematic review of the literature. The steps outlined in the guidelines for this structured analysis include review planning, review implementation, and review documentation (Fig. 1).

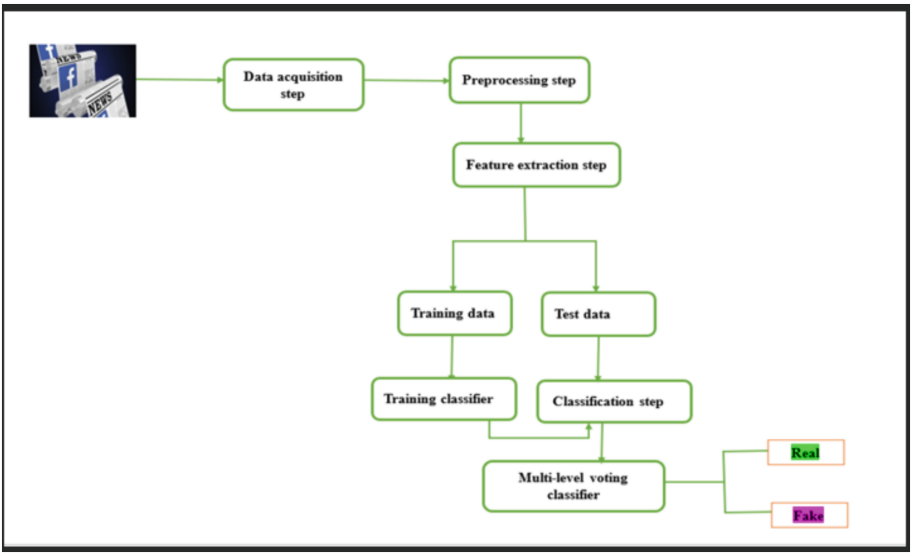


Fig. 1. Depicts the overall architecture of fake news detection.

3.1 Review Plan

The ongoing analysis of the existing body of literature aims to thoroughly examine the predominant issues and gaps in ensuring the security of systems. This examination will delve into a wide array of systems and contexts, seeking to investigate the various demands, measurements, structures, and methodologies involved. The primary goal of this analysis is to provide a systematic categorization of false news detection and to critically assess important fundamental principles, advancements in technology, trends in research, and future directions.

3.2 Establishing Review Protocols

A review protocol is developed after a comprehensive analysis of the current methodology utilized in the systematic review, followed by extensive consultations with esteemed professionals in the relevant field of study. This protocol is designed to include a set of guidelines and criteria for the assessment of the quality of studies, the selection of research subjects, an outlined approach for executing a thorough search, various potential sources to be considered, specific criteria for the selection process, as well as a detailed procedure for the identification and inclusion of relevant materials. Furthermore, it encompasses detailed specifications of the extraction and synthesis of data, along with essential elements related to the planning and management of the review, such as the development of a project schedule and strategies for the dissemination of the research findings.

3.3 Examine and Assess the Guidelines

- Are the search terms relevant and in line with the research questions?
- Will the data extraction points adequately address the research questions?
- Is the analysis strategy suitable for accomplishing the goals of the review?

3.4 Formulate Inquiry Queries

For these reviews, the following study issues were investigated after careful consideration, discussion, and reformulation:

RQ. 1. How do existing deep learning models detect fake news in Afan Oromo compared to other languages?

RQ.2. What are the primary challenges and limits of using deep learning to detect fake news in Afan Oromo?

RQ.3. How can the evaluation of deep learning-based models for Afan Oromo fake news on social media improve research, boost media literacy, and empower local communities to confront misinformation?

3.5 Reliability in Inclusion Decisions

Every digital source is assigned to two analysts to ensure the consistency of the consideration decisions. Two analysts evaluate each article following the incorporation and avoidance criteria. To determine if scholars agree on consideration or prohibition, a list of distributions with consideration and prohibition criteria is maintained. Any discrepancies or flaws in any document were addressed during the group's common assembly and resolved through discussion and professional help (Table 4).

Table 1. Shows the related work in this field by various researchers.

Authors & Year	Research Title	Key Findings, Contributions, and Conclusion	Remarks on the suggested research that is subject to criticism
Daraje Kaba Gurmessa (2020a) [17]	Afan Oromo Fake News Detection Using Natural Language Processing and Passive-Aggressive	The researcher has focused mostly on how a large number of corpora are necessary to construct a text content-based fake news detecting system and in light of this, the study designated the Afan Oromo news corpus	The research study is pertinent to the research proposal, but the researcher concentrated only on, constructing a text-content-based Afan Oromo fake news detecting system
G. Bharath, K J Manikanta, G Bhanu Prakash, Sumathi, P. Chinnasamy (2021) [18]	Detecting Fake News Using Machine Learning Algorithm	The authors understand how to conceptualize accuracy checks and taking into account automatically fashioned news distinguishing evidence in Twitter datasets, they present a model for understanding manufactured news messages from Twitter tweets	It is good for research work however; the researchers are emphasizing only Twitter datasets
HundaolBedada Regesa (2020) [19]	Detection And Classification of Afan Oromo Fake News on social media: - Using A Machine Learning Approach	They developed datasets for the detection and classification of Afan Oromo fake news stories, identified the algorithm utilized for this study, trained and tested their model, and assessed the effectiveness of the model as their primary contribution	The researchers developed datasets for the detection and classification of Afan Oromo fake news but for this investigation, the researcher only gathered a modest number of datasets in all, 523 data were obtained
Uma Sharma, Sidarth Saran, Shankar M. Patil (2020) [20]	Fake News Detection Using Machine Learning Algorithms	The main aim of the researcher is to conduct binary categorization of diverse news items available online using principles from Artificial Intelligence, Natural Language Processing, and Machine Learning	The authors somehow don't specify the algorithm they employed, merely stating that they used Artificial Intelligence, Natural Language Processing, and Machine Learning

(*continued*)

Table 1. (*continued*)

Authors & Year	Research Title	Key Findings, Contributions, and Conclusion	Remarks on the suggested research that is subject to criticism
Ali Jarrahi1 & Leila Safari1 (2022) [21]	Evaluating the effectiveness of publishers' features in fake news detection on social media	Some of the significance of the research work is that credibility, Influence, Sociality, Validity, and Lifetime were introduced as key features for news publishers on social media. The authors proposed CreditRank, an algorithm for assessing publishers' credibility on social networks, and also proposed FR-Detect, a high-accuracy multi-modal system for detecting false news utilizing user-related and content-related features	The researchers investigate evaluating the effectiveness of the researcher's publishers' features in fake news but they focus on only two kinds of data sources to evaluate the researcher namely Gossip Cop (news related to celebrities) and PolitiFact (political news), as well as they used only Twitter dataset
Z Khanam, B N Alwasel, H Sirafi, and M Rashid (2021) [22]	Fake News Detection Using Machine Learning Approaches	This paper's research focuses on detecting fake news through two stages: characterizations and disclosure. In the first step, social media is used to expose the fundamental concepts and principles of fake news. During the discovery step, existing strategies for detecting fake news utilizing various supervised learning algorithms are reviewed	The authors focus only on political fake news datasets. In addition, they used a dataset from Kaggle. They did not try to collect by themselves

(*continued*)

Table 1. (*continued*)

Authors & Year	Research Title	Key Findings, Contributions, and Conclusion	Remarks on the suggested research that is subject to criticism
Muhammad Pervez Akhter, Jiangbin Zheng, Farkhanda Afzal, Hui Lin, Saleem Riaz and Atif Mehmood (2021) [23]	Supervised ensemble learning methods toward automatically filtering Urdu fake news within social media	This topic is investigated in this study in the Urdu language They are making three contributions. For the false news identification tasks, first, they create an annotated corpus of Urdu news articles. Second, they investigate three distinct machine-learning methods for detecting fake news. Third, they apply five ensemble learning approaches to ensemble the predictions of the base predictors to improve the overall performance of the false news detection system	Under the findings of this research work the authors developed the dataset and focused on only the Urdu language. Various limitations of this study should be addressed in future research. In the English-to-Urdu translation, they used the online Google translation API, and they believe the translation accuracy is not perfect
Md Rafiqul Islam Shaowu Liu Xianzhi Wang Guandong Xu (2020) [24]	Deep learning for misinformation detection on online social networks: A survey and new perspectives	They give an in-depth examination of automated misinformation detection on (i) inaccurate information, (ii) rumors, (iii) spam, (iv) fake news, and (v) disinformation Researchers propose a cutting-edge thorough review of existing MID challenges, solutions, and validation in online social networks using multiple DL approaches They assess the key strengths and limits of the existing diverse methodologies and describe state-of-the-art DL as a new methodology on vast social network data to uncover recent and future trends in MID research They present some open issues that contribute to this new and intriguing subject based on machine learning techniques	-

(*continued*)

Table 1. (*continued*)

Authors & Year	Research Title	Key Findings, Contributions, and Conclusion	Remarks on the suggested research that is subject to criticism
Gurmessa, D. K (2020b) [14]	Afan Oromo Text Content-Based Fake News Detection using Multinomial Naive Bayes	The author applied the term frequency-inverse document frequency (TF-IDF) of unigrams and bi-grams and used Facebook as the news article's source. Based on their data size, they were able to obtain good outcomes with the best f1 score. Based on their data size, they were able to obtain good outcomes with the best f1 score. Although the accuracy acquired is good, the information from the data set site's source is not entirely reliable, and the highest accuracy might have been attained by taking unreliable news into account	Even though the researcher proposed and applied good outcomes the information from the data set site's source is not entirely reliable, and the highest accuracy might have been attained by taking unreliable news into account

Table 2. Provides details on the free datasets and any supporting materials

Dataset	Modality	Size	Label	Type	Public availability
Fake news [14]	Text	752	Fake and real	Articles	The datasets generated during and analyzed during the current study are not publicly available but are available from the corresponding author on reasonable request
Multi-modal features [25]	Text, image,	5K	Both fake and real	Articles	Yes
Fake news [26]	Text, image, sound	31213	Fake and real	Articles	Yes

(*continued*)

Table 2. (*continued*)

Dataset	Modality	Size	Label	Type	Public availability
COVID-19 news [12]	Multimodal	4300	Fake and real	Articles	Yes
Fake news [13]	Text	7000	Fake and real	Thesis	No
Fake news [19]	Text	646	Fake and real	Thesis	No
Fake news [29]	Text	23KB	Both fake and	Article	Yes
Political fake news [21]	Text	15,751	Fake and real	Article	Yes
News text [30]	Text	824	Fake and real	Articles	No
News [31]	Image, text	33,499	Fake and real		Yes

Table 3. Research goals and inspirations

Id	Approach Questions	Motivations
RQ1	How do existing deep learning models detect fake news in Afan Oromo compared to other languages?	This investigation points to the numerous techniques and approaches that have been looked into and talked about within the writing to examine fake news. Finding and surveying germane investigative papers, articles, and distributions is what the systematic literature survey involves to urge an understanding of the methods and information that are as of now known within the subject of fake news
RQ2	What are the primary challenges and limits of using deep learning to detect fake news in Afan Oromo?	By investigating the primary challenges and limitations of using deep learning to detect fake news in Afan Oromo, researchers can gain valuable insights into the specific challenges that must be overcome to improve the effectiveness and reliability of detection models in this language and cultural context

(*continued*)

Table 3. (*continued*)

Id	Approach Questions	Motivations
RQ3	How can the evaluation of deep learning-based models for Afan Oromo fake news on social media improve research, boost media literacy, and empower local communities to confront misinformation?	The assessment of deep learning-based models for identifying false news in Afan Oromo can help to advance research in natural language processing and machine learning, increase media literacy, empower local communities, and promote collaboration. By offering tools for detecting and combating false news, these models may increase awareness about the prevalence of disinformation on social media, empower individuals to make informed decisions and build a culture of truth-seeking and responsibility. Furthermore, this study can encourage collaboration among researchers, policymakers, media organizations, and community stakeholders to address the difficulties posed by disinformation on social media, enabling collective action and capacity-building activities

Table 4. Inclusion and exclusion criteria for selection

Category	Criteria for Inclusion and Exclusion
Inclusion	• In the research, Toward the Development of Deep-Learning-Based Models for Afaan Oromo Fake News Detection and Classification on Social Media Networks, we address at least one issue about the requirements for fake news detection and classification • The articles are on the subjects of deep learning and natural language processing • Papers were peer-reviewed in journals and at conferences
Exclusion	• The papers are not conducted in the context of the Development of Deep-Learning-Based Models for Afaan Oromo Fake News Detection and Classification on Social Media Networks • Articles for training and experimenting on Deep Learning-Based Models and Fake News Detection and Classification on Social Media Networks • Journals are non-peer-reviewed then don't relate to an abstract of a whole volume or a journalistic

4 Result and Finding

Evaluating the output of a machine-learning model is a critical stage in a predictive modeling pipeline. An evaluation matrix is a critical tool for planning and organizing an evaluation, and the confusion matrix displays a summary of model performance on the testing dataset [32]. The number of labels in most classification problems is fixed, so determining the score for each class and the loss from the ground truth is straightforward. In the case of NLP, even though the output format is predetermined, dimensions cannot be chosen, as a single remark can be presented in a variety of ways without changing its intent or meaning. If we are attempting to solve two problems with one model, evaluation measures are critical for assessing its performance. Various evaluation measures will be utilized to assess the effectiveness of algorithms for the false news identification problem.

The majority of current methods view the fake news problem as a classification issue that seeks to determine if a news story is false or not [33]. The Afaan Oromo false news dataset was used in this study to test the performance of the machine learning models, and the confusion matrix was used to represent the classification's four possible outcomes. In our instance:

True Positives (TP): The news item has been deemed false and is fake.

False Positives (FP): The news item has been mistakenly identified as true despite being false.

False Negatives (FN): The news item is accurate but labeled as phony.

The news item has been evaluated as true and is true, according to the True Negative (TN).

We extract the following measures, which will be used to assess our models, from those measurements.

Accuracy is measured by the ratio of accurate predictions to the total amount of data sets.

Precision: is the percentage of correctly identified positives that we correct.

4.1 Accuracy

The accuracy score, also known as the classification accuracy rating, is calculated as the proportion of correct predictions to total predictions generated by the model. The given formula in Eq. (1) might illustrate the accuracy.

$$\mathrm{A} = \frac{True\ Positive + True\ Negative}{Total\ Number\ of\ Predictitions} \tag{1}$$

4.2 Precision

Precision (P) is defined as the number of actual positive findings divided by the total number of positive results, including incorrectly recognized ones. The precision can be computed using Eq. (2).

$$P = \frac{True\ Positive}{Positive + False\ Positive} \tag{2}$$

4.3 Recall

When the total number of samples that should have been identified as positive is used to divide, the number of true positive results is referred to as recall (R). The recall can be computed using Eq. (3)

$$R = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{3}$$

5 Challenges and Research Direction

Natural Language Processing (NLP) applications are expanding at a rapid pace, and new issues are emerging as a result. Contextual words and phrases in the language, dealing with synonyms, different levels of complexity in language, homonyms, sentences using sarcasm and irony, language containing informal phrases, expressions, idioms, and culture-specific lingo, misspelled or misused words, and predicting the writer's intention from a specific domain, geographic area are some of the common challenges [34]. These issues must be addressed for a broad application of NLP, as the models may operate well in a certain domain or geographic area, but for a broad use, such challenges must be addressed [35].

Furthermore, misspelled or misused phrases might cause issues, even though auto-correct and grammar correction software has greatly improved due to continual advancements in the field. The most essential characteristics of the term's problems, false, news, photos, and features are that there is a need for models for all people rather than specialist linguistic skills and technology to detect fake news.

Deep learning-based methods outperform conventional methods in terms of accuracy, although there is still room for further progress and inquiry. The selection of features and classifiers has a significant impact on the model's efficiency, and past research did not place a high importance on the selection of features and classifiers. While news content and headline elements are extensively employed in fake news identification, other features such as user behaviors, user profiles, and social network behaviors must be investigated. Political or religious bias in profile variables, as well as lexical, syntactic, and statistical-based factors, can boost detection rates. There are few propagation-based studies in this domain, and meta-data and other information can improve resilience and reduce the noise of a single textual claim.

An inquiry area for building a better and more robust system will be an assessment with video and image features. Fake news detection methods that learn in real-time from newly appearing online content could improve detection outcomes. More data for a larger quantity of fake news should be released, as data scarcity is a big issue in fake news categorization. Ensemble approaches outperform a basic classifier, and GRU models outperform an LSTM. Transformers have supplanted RNN models such as LSTM as the model of choice for NLP tasks, and GPT can be leveraged by fine-tuning false news detection jobs. There is always room for improvement and research, although many studies have been done on the detection of fake news. In terms of identifying false news, we emphasize difficulties as well as some original research topics. Although

Deep Learning-based methods are more accurate than conventional methods, there is still room to improve their acceptability.

The choice of features and classifiers has a significant impact on the model's effectiveness. The choice of features and classifiers was not given much attention in earlier studies. Researchers should concentrate on figuring out which classifier is best for specific traits [35].

The usage of sequence models (RNNs) is required for the long textual features, although only a few research papers have done so. We believe that studies that concentrate on the selection of features and classifiers might potentially improve performance.

The feature engineering concept is not common in deep learning-based studies. News content and headline features are the widely used features in fake news detection, but several other features such as user behavior [36], user profile, and social network behavior need to be explored. Political or religious bias in profile features and lexical, syntactic, and statistical-based features can increase the detection rate. A fusion of deeply hidden text features with other statistical features may result in a better outcome. There isn't much research in this field that uses propagation [34]. A piece of information that hasn't been fully used for false news identification is network-based patterns of news transmission [37]. As a result, we advise taking news dissemination into account when identifying bogus news. Although they must be used carefully, meta-data and supplementary information can strengthen a textual claim and lessen its noise.

Studies solely examined text data to detect false news, yet fake news is produced using sophisticated techniques and purposely manipulated text or images [38]. Image features have only been used in a few research [39, 40]. As a result, we advise using visual data (videos and photographs). To create a stronger and more reliable system, an inquiry into video and image features will be conducted. There aren't many studies in this field that combine features [41]. Combining data from many sources could be very helpful in figuring out whether Internet articles are fraudulent [38]. We recommend combining subsequent pre-trained word embeddings with multi-model-based methodologies. The detection of bogus news may be significantly impacted by numerous other hidden aspects. As a result, we urge academics to look into hidden aspects.

Real-time learning from newly published online articles by fake news detection programs may improve detection outcomes. Using a transfer-learning strategy to train a neural network using online data streams is another intriguing future endeavor.

Since the absence of data is the primary issue in the classification of fake news, more data for a larger number of false news items should be made public. More training data are thought to enhance model performance. Publicly accessible datasets with a news content focus are available. Datasets based on various textual properties, however, are few. Research that makes use of additional textual features is therefore rare.

Results are improved by utilizing an ensemble technique rather than a straightforward classifier [42]. Better results can be obtained by building an ensemble model using DL and ML methods, in which an LSTM can recognize the original article while feeding auxiliary characteristics through a second model [43]. An LSTM performs worse than a more straightforward GRU model. Consequently, we advise combining GRU and CNNs to urge the leading result.

Using, many studies have attained great accuracy.

Ensemble models, CNN, and LSTM [36, 44]. In this domain, SeqGAN and Deep Belief Network (DBN) were not investigated. We invite scientists to test out these models.

For NLP tasks, transformers have taken the place of RNN models like LSTM. While Generative Pre-Trained Transformer (GPT) has not been applied in this field, BERT has been used to identify bogus news. We advise employing GPT by honing tasks for bogus news identification.

Current algorithms make important decisions without giving detailed details about the reasons behind particular judgments, projections, suggestions, or actions.

6 Conclusion

This study explores fake news detection, focusing on its various types, features, and distinguishing characteristics. It provides an extensive taxonomy of fake news detection models and compares traditional machine learning methods against contemporary deep learning techniques. The study focuses on three categories of deceptive information: liars, fake news publications, and a selection of corpus datasets. The findings are valuable for future investigations aimed at enhancing the identification of fake news and developing new models or tools for timely detection. However, the analysis primarily focuses on textual data, and future research could benefit from analyzing image data alongside textual information. The review critically examines various methods for identifying fake news, including natural language processing (NLP) and advanced deep learning (DL) strategies. It explores diverse NLP techniques and deep learning architectures, examining various assessment metrics and outlining potential future research trajectories. The ongoing evolution and development of innovative deep learning network architectures significantly reduce the risk of obtaining erroneous results when implementing deep learning-based models. The findings of this review are promising and reliable due to the ongoing discourse surrounding this pressing societal challenge. The study provides an exhaustive examination of the mechanisms underlying the identification of false news, emphasizing its defining characteristics, critical features, comprehensive taxonomy, various types of data, categorization of false news, and detection methodologies employed to unearth fake news.

Declarations. Funding: Authors declare no funding for this research.
Competing Interests: The authors declare that they have no competing interests.
Conflicts of Interest: The authors declare that they have no conflict of interest

References

1. Virdi, P.: Fake accounts detection in facebook (2017). https://doi.org/10.1109/ISEASP.2017.7976996
2. Baarir, N.F., Djeffal, A.: Fake news detection using machine learning. In: IEEE 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH), pp. 125–130 (2021). https://doi.org/10.1109/IHSH51661.2021.9378748
3. Bugade, P., Sarode, P., Pimple, T.: Fake news detection using machine learning. Int. Res. J. Eng. Technol. **08**(04), 4024–4027 (2021). https://www.irjet.net/archives/V8/i4/IRJET-V8I4773.pdf

4. fao, "fao.org,". Secondary sources of information (2023). https://www.fao.org/3/w3241e/w3241e03.htm. Accessed 09 Jan 2023
5. Patil, J.S.R.: Fake news detection using machine learning. In: IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC), pp. 1–5 (2020). https://doi.org/10.1109/iSSSC50941.2020.9358890
6. Mandical, R.R., Mamatha, N., Shivakumar, N., Monica, R., Krishna, A.N.: Identification of fake news using machine learning, pp. 1–6. IEEE (2020). https://doi.org/10.1109/CONECCT50063.2020.9198610
7. De Paor, S., Heravi, B.: Information literacy and fake news: how the field of librarianship can help combat the epidemic of fake news. ELSEVIER J. Acad. Librarianship 1–8 (2020). https://doi.org/10.1016/j.acalib.2020.102218
8. Amri, S., Aïmeur, E., Brassard, G.: Fake news, disinformation and misinformation in social media: a review. Springer Social Network Analysis and Mining, pp. 1–36 (2023). https://doi.org/10.1007/s13278-023-01028-5
9. Olan, F., et al.: Fake news on social media: the impact on society Springer Information Systems Frontiers, pp. 1–16 (2022). https://doi.org/10.1007/s10796-022-10242-z
10. Nistor, A., Zadobrischi, E.: The influence of fake news on social media: analysis and verification of web content during the COVID-19 pandemic by advanced machine learning methods and natural language processing. MDPI (Multidisciplinary Digital Publishing Institute) 0nline:10466 (2022). https://doi.org/10.3390/su141710466
11. Mohawesh, R., Maqsood, S., Althebyan, Q.: Multilingual deep learning framework for fake news detection using capsule neural network. Springer Journal of Intelligent Information Systems, pp. 1–17 (2023). https://doi.org/10.1007/s10844-023-00788-y
12. Alghamdi, J., Lin, Y., Luoa, S.: Towards COVID-19 fake news detection using transformer-based models. Elsevier Knowledge-Based Systems, pp. 1–12 (2023). https://doi.org/10.1016/j.knosys.2023.110642
13. Waldesebat, A.: Fake news detection on social media by using deep learning for afaan oromo language. Ambo University (2021). https://www.scribd.com/document/514050866/Fake-news-detection-on-social-media-by-using-deep-learning-approach-for-afaan-oromoo-language
14. Gurmessa, D.K.: Afaan oromo text content-based fake news detection using multinomial naive bayes. Int. J. Innov. Manage. Sci. Eng. **01**, 26–37 (2020b).https://www.researchgate.net/publication/347508623_Afaan_Oromo_Text_Content-Based_Fake_News_Detection_using_Multinomial_Naive_Bayes
15. Abdulrahman, A., Baykara, M.: Fake news detection using machine learning and deep learning algorism. Third International Conference on Advanced Science and Engineering (ICOASE2020), pp. 18–23 IEEE (2020. https://doi.org/10.1109/ICOASE51841.2020.9436605
16. Aphiwongsophon, S., Chongstitvatana, P.: Detecting fake news with machine learning method. In: IEEE: 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, pp. 528–531 (2018). https://doi.org/10.1109/ECTICon.2018.8620051
17. Gurmessa, D.K.: Afaan oromo fake news detection using natural language processing and passive-aggressive. UIJRT | United Int. J. Res. Technol. **02**, 2582–6832 (2020). https://www.researchgate.net/publication/347931596
18. Bharath, G., Manikanta, K.J., Bhanu Prakash, G., Sumathi, R., Chinnasamy, P.: Detecting fake news using machine learning algorithm. IEEE International Conference on Computer Communication and Informatics (ICCCI -2021), pp. 1–5 (2021). https://doi.org/10.1109/ICCCI50826.2021.9402470

19. Regesa, H.B.: Detection and classification of afaan oromo fake news on social media: - using a machine learning approach. Ambo University, Woliso Campus (2020). https://www.scribd.com/document/488591181/Detection-and-Classification-of-Afaan-Oromo-Fake-News-on-Social-Media-Using-a-Machine-Learning-Approach
20. Sharma, U., Saran, S., Patil, S.M.: Fake news detection using machine learning algorithm. Int. J. Creat. Res. Thoughts **08**(06), 1394–1402 (2020). https://www.researchgate.net/publication/343916946
21. Jarrahi, A., Safari, L.: Evaluating the effectiveness of publishers' features in fake news detection on social media. Multimedia Tools and Applications Springer, pp. 1–27 (2022). https://doi.org/10.1007/s11042-022-12668-8
22. Khanam, Z., Alwasel, B.N., Sirafi, H., Rashid, M.: Fake news detection using machine learning approach. In: ASCI-2020 IOP Publishing Conference Series: Materials Science and Engineering, pp. 1–14 (2021). https://doi.org/10.1088/1757-899X/1099/1/012040
23. Akhter, M.P., Zheng, J., Afzal, F.: Supervised ensemble learning methods towards automatically filtering Urdu fake. PeerJ Comput. Sci. 1–25 (2021). https://doi.org/10.7717/peerj-cs.425
24. Islam, M.R., Liu, S., Wang, X., Xu, G.: Deep learning for misinformation detection on online social networks. Springer Social Network Analysis and Mining, pp. 1–20 (2020). https://doi.org/10.1007/s13278-020-00696-x
25. Garg, D.K.S.S.: IFND: a benchmark dataset for fake news detection. Springer, pp. 1–21 (2023). https://doi.org/10.1007/s40747-021-00552-1
26. Jolly, S.K.A.: Fake news detection in different applications: state-of-the-art and future directions. Research Square, pp. 1–34 (2022). https://doi.org/10.21203/rs.3.rs-2024922/v1
27. Himdi, H., Weir, G., Assiri, F., Al-Barhamtoshy, H.: Arabic fake news detection based on textual analysis. Springer Arabian Journal for Science and Engineering, pp. 1–17 (2022). https://doi.org/10.1007/s13369-021-06449-y
28. Tashtoush, Y., Alrababah, B., Darwish, O., Maabreh, M., Alsaedi, N.: A deep learning framework for detection of COVID-19 fake news on social media platforms. MDPI data, pp. 1–18 (2022). https://doi.org/10.3390/data7050065
29. Dey, A., Reddy, H., Dey, M., Sinha, N.: Detection of fake accounts in instagram using machine learning. Int. J. Comput. Sci. Inform. Technol. **11**, 1–8 (2019). https://doi.org/10.5121/ijcsit.2019.11507
30. Jimalo, K.M., Babu, R., Assabie, Y.: Afaan oromo news text categorization using decision tree classifier and support vector machine: a machine learning approach. Int. J. Comput. Trends Technol. **47**, 29–41 (2017). https://ijcttjournal.org/2017/Volume47/number-1/IJCTT-V47P101.pdf
31. Shah, P.V.: Multimodal fake news detection using a Cultural Algorithm with situational and normative knowledge Electronic Theses and Dissertations. IEEE World Congress on Computational Intelligence (2020). https://doi.org/10.1109/CEC48606.2020.9185643
32. Singh, D.K., Koli, A., Sukhdev, K.K.: Natural language processing: state of the art, current trends and challenges. Springer Multimedia Tools and Applications, pp. 3713–3744 (2023). https://doi.org/10.1007/s11042-022-13428-4
33. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. arxiv.org Computer Science > Social and Information Networks (2017). https://arxiv.org/abs/1708.01967
34. Wu, L., Liu, H.: Tracing fake-news footprints: characterizing social media messages by how they propagate. In: WSDM '18: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 637–645 (2018). https://doi.org/10.1145/3159652.3159677

35. Bhanu, P.S., Mary Saira, S.: Ubcadet: detection of compromised accounts in Twitter based on user behavioral profiling," Multimedia Tools and Applications. Springer Multimedia Tools and Applications, pp. 19349–19385 (2020). https://doi.org/10.1007/s11042-020-08721-z
36. Ding, J., Hu, Y., Chang, H.: Bert-based mental model, a better fake news detector. In: Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence, pp. 396–400 (2020). https://login.research4life.org/tacsgr1_org/10.1145/3404555.3404607
37. Zafarani, X.Z.R.: Fake news detection: an interdisciplinary research. In: Companion Proceedings of The 2019 World Wide Web Conference, WWW '19: Companion Proceedings of The 2019 World Wide Web Conference, p. 1292 (2019). https://doi.org/10.1145/3308560.3316476
38. Giachanou, A., Zhang, G., Rosso, P.: Multimodal multi-image fake news detection. In: 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), pp. 647–654 (2020)
39. Mangal, D., Sharma, D.K.: Fake news detection with the integration of embedded text cues and image features. In: 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), pp. 68–72 (2020). https://doi.org/10.1109/ICRITO48877.2020.9197817
40. Qi, P., Cao, J., Yang, T., Guo, J., Li, J.: Exploiting multi-domain visual information for fake news detection. In: 2019 IEEE International Conference on Data Mining (ICDM), pp. 518–527 (2019). https://doi.org/10.1109/ICDM.2019.00062
41. Kumari, R., Ekbal, A.: Amfb: attention-based multimodal factorized bilinear pooling for multimodal fake news detection," Expert Systems with Applications. Expert Syst. Appl. Int. J. **184** (2021). https://doi.org/10.1016/j.eswa.2021.115412
42. Jadhav, S.S., Thepade, S.D.: Fake news identification and classification using dssm and improved recurrent neural network classifier. Appl. Artific. Intell. **33**, 1058–1068 (2019). https://doi.org/10.1080/08839514.2019.1661579
43. Deepak, S., Chitturi, B.: Deep neural approach to fake news identification. In: Procedia Computer Science, International Conference on Computational Intelligence and Data Science, vol. 167, pp. 2236–2243 (2020). https://doi.org/10.1016/j.procs.2020.03.276
44. Umer, M., Imtiaz, Z., Ullah, S.: Mehmood, A., Choi, G.S., On, B.W.: Fake news stance detection using deep learning architecture (cnn-lstm). IEEE Access **8** (2020). https://doi.org/10.1109/ACCESS.2020.3019735

Integrating AI and Block Chain for Enhanced Food Traceability and Safety in Restaurants: A Transformative Approach

Sachin Yele(✉) and Ratnesh Litoriya

Medi-Caps University, Indore, India
sachin.yele27@gmail.com

Abstract. In an effort to improve restaurant food safety and traceability, this research aims to investigate the revolutionary possibilities of integrating block chain technology with Artificial Intelligence (AI). Innovative approaches to food security are urgently needed as the Fourth Industrial Revolution moves forward, bringing together physical and digital technology. One promising approach is precision agriculture, which combines traditional knowledge with modern technological advancements. Improving food traceability, increasing transparency across the supply chain, and protecting consumers are all goals of the proposed effort, which uses AI and block chain to evaluate these potential benefits. The suggested system aims to cut down on cases of food poisoning and improve customer safety by using data collected in real-time from the food supply chain. Research shows that the restaurant business may greatly benefit from AI-driven insights and block chain's strong record-keeping capabilities when it comes to food traceability and safety.

Keywords: Block chain · AI · Supply chain · Precision Agriculture

1 Introduction

As the Fourth Industrial Revolution has sped digital technology breakthroughs across many industries and transformed human civilization in the twenty-first century, a "cyber-physical system" that blends automation and cyber technologies has evolved as a characteristic. A few examples of recent technological developments are autonomous vehicles, internet-enabled robots, microchips, cloud computing, block chain, nanotechnology, biotechnology, cloud analytics, photonics, and internet connectivity. The Fourth Industrial Revolution has the potential to impact several areas of people's life, not only the industry.

The fourth industrial revolution is based on the idea of integrating physical industrial resources with digital technological advancements. To overcome upstream and downstream difficulties, digital transformation is essential in the agriculture industry, the primary provider of food for many people's lives. The agro-industry may be transformed into an effective upstream-to-downstream production value chain business model by integrating digital technology with potential resources of local knowledge in agricultural management. The fruitful results of combining technological advancements with

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 59–70, 2025.
https://doi.org/10.1007/978-3-031-86296-0_5

agricultural progress 4.0 affects the community's capacity to earn more money and get their hands on basic foods. A key component of Indonesia's agricultural growth is precision agriculture, which emphasizes the use of digitalization technology to agricultural production systems and the appropriate utilization of resources. One solution to the problem of meeting global food demands through the development of new, more efficient production methods is the use of block chain and artificial intelligence (AI) in agriculture. Data can be recorded in real-time throughout the food supply chain thanks to block chain technology and AI, which also allows for transparency and information tracking [1].

This paper looks at how we may greatly benefit from fusing block chain technology with artificial intelligence. These two technology working together can produce really good results.

In conclusion, this paper makes the case that restaurants should adopt block chain technology and AI as a revolutionary way to ensure food safety and improve food traceability. The study's findings clearly demonstrate the need for technology improvements in restaurant inspection in order to guarantee food safety and traceability.

1.1 Motivation

AI and Block chain are both extremely prominent technology in their respective fields. When combined, they can bring good answers to the difficulties facing India, where lakhs of people are dying because of food-related ailments. Block chain technology employs a distributed ledger, whereas AI develops knowledge-based methods. The combination of these two can result in a very effective model that is essential for everyone. Implementing this strategy in India's food security system might help us address critical issues.

1.2 Organization of Paper

The following is the outline for the remainder of the paper. The fundamentals of blockchain, artificial intelligence are covered in Sect. 2. The Literature review done in Sect. 3. The methodlogy for th work described in Sect. 4 and also shows the applications like in the food processing industry, with an emphasis on data processing aspects such as security, visualization, analysis. Section 5 brings the result and discussion, while Sect. 6 provides an overview of future trends and problems, lastly Sect. 7 conclude the article.

2 Artificial Intelligent and Block Chain: State of the Art

2.1 Artificial Intelligent (AI)

AI, which refers to "the simulation of human intelligence in machines that is programmed to think and act like humans," is a popular area of study in the field of computer science. Included in its vast scope are ideas like deep learning, machine learning (ML), and many more. There have been several phases in the evolution of AI, including (i) rules based on knowledge, (ii) patterns using ML, and (iii) automation utilizing deep learning.

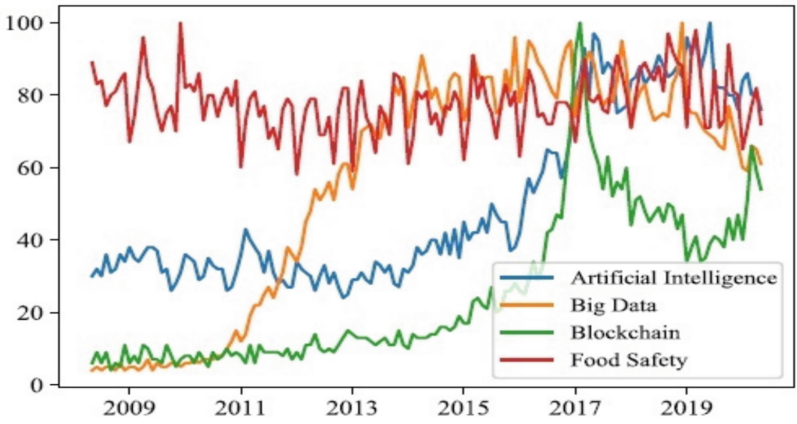


Fig. 1. Relative World Wide Traffic Data for Food Safety [2]

Human specialists establish knowledge-based rules in the first stage of AI, which is subsequently utilized to identify rules or make judgments [2].

From 2009 to 2019, the Fig. 1 shows the developments in four important technical and conceptual areas: AI, Big Data, Blockchain, and Food Safety. The x-axis shows the data throughout time, while the y-axis shows how often or popular each issue is. There has been a slow but steady uptick in interest in artificial intelligence (blue line), especially after 2015, and a meteoric surge in popularity by 2019. The orange line represents big data, which begins to gain popularity about 2011, reaches its peak in 2014 and 2015, and then stays pretty high and constant after that. Although it demonstrates indications of oscillation near 2019, blockchain (green line), which was mostly unknown until 2015, undergoes a tremendous spike in popularity beginning in 2017. Food Safety, on the other hand, shows a more irregular trend, with large swings throughout the years (red line). Despite its prominence in 2009, interest in the subject seems to have leveled out and then spiked again in the middle of the 2010s. While more established subjects, such as food safety, show less stability over time, the chart shows that new technologies, such as AI and blockchain, are rising rapidly [3].

2.2 Blockchain

The term "blockchain" can refer to a distributed ledger that records all transactions and other actions taken by all parties engaged in a certain transaction. From individual producers and processors to distributors and consumers, everyone in the supply chain stands to gain from blockchain technology's use [4, 5].

1. Distributed ledger technology, or blockchain, allows users to store and access data independently of one another.
2. Without the need for an intermediary, any peer or authority with a legitimate identity may conduct transactions and verify information directly with other parties.
3. Blockchains use cryptography to secure transaction records and a consensus data verification method.

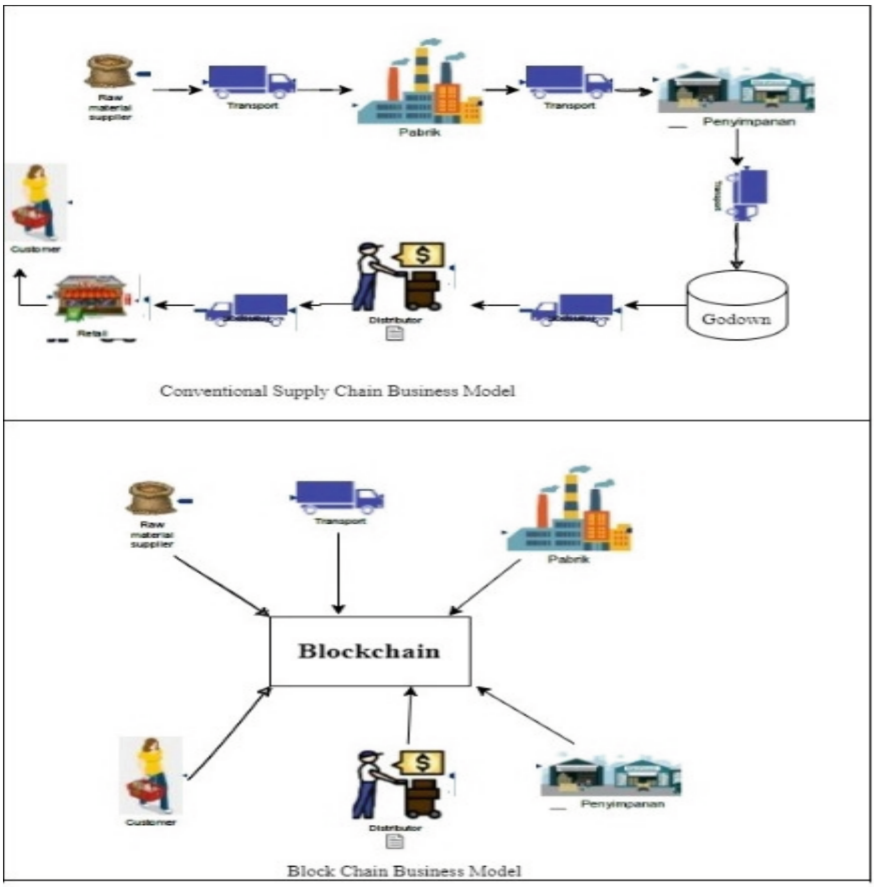


Fig. 2. Differences between conventional supply chain business models and blockchain business models

A traditional supply chain business model is shown in at the top of the Fig. 2 [6], while a blockchain-based model is shown at the bottom. Transporting raw materials to manufacturers ("Pabrik"), then to storage facilities ("Penyimpanan"), and last to distributors and retailers, the traditional model of the supply chain depicts a linear progression culminating in the consumer. Verification and coordination are necessary at each stage of this model's many processes. The blockchain approach, on the other hand, streamlines everything by linking all parties directly through a decentralized blockchain network, including producers, warehouses, distributors, and consumers. By recording all transactions on the blockchain in real-time, this makes the supply chain more efficient and secure, streamlines the movement of products and information, and promotes transparency. It also removes intermediaries.

3 Literature Review

Arkeman et al. [1] provide information about the system that customer may examine their purchase history and the current status of their products in a matter of seconds with the use of digital systems that are based on artificial intelligence (AI) and blockchain technology. An improved system, referred to as a smart blockchain, will be the result of integrating blockchain technology with AI.

Zhou et al. [2] provides an overview of AI and Blockchain technologies and how they may be applied to the field of food safety.First, the fundamental ideas behind these technologies are laid out. After that, we take a look at the data-centric uses of these technologies in food safety applications. Last but not least, we go over some of the problems that can arise in the future when using AI, big data, and blockchain.

Tian et al. [4] The current state of knowledge on the usage of blockchain technology in infrastructure financing and the function of tokenization within the framework of public-private partnerships and project financing is enhanced by this study.

YU et al. [7] In order to test the practicality of the proposed system, author build a version of the quality monitoring system for peach juice manufacturing using the Remix IDE and the Ethereum platform.

Tsoukas et al. [8] seeks to secure all components of food supply chain monitoring systems in order to increase their transparency using TinyML's nascent technology with Blockchain.

Min Chen et al. [9] prvide the concept that Using machine learning, the architecture can first determine if there has been an issue with this batch of food during the transaction, and if so, it can implement more thorough anomaly detections using the supplier's equipment to pinpoint exactly when the problem occurred.

Gómez et al. [10] This study contributes to the field of methodology by providing concrete examples of how systematic literature reviews might be conducted using machine learning techniques including Blockchain Technology in food industry.

Mohan Raparthi [11] offers a thorough examination of ML-based blockchain supply chain management, with an emphasis on decentralized solutions for transparency and traceability.

Jiang and Lei [12] provide information about the work that refines the task model and applies deep reinforcement learning's deep Q network technique to accomplish many optimization goals specified by the task scheduling issue, to keep product quality high and improve traceability of items.

CHEN et al. [13] simulate the usefulness of the proposed blockchain-based infrastructure and the DR-SCM approach is demonstrated through experiments done in various ASC scenarios for reliable product traceability.

Nandakumar et al. [14] propose a blockchain-based food traceability system that is powered by machine learning (MBCFTS) to provide betterment of FSC using blockchain-based FSC framework and second to integrate blockchain and cloud technology.

Lin et al. [15] Combining artificial intelligence with blockchain technology enhanced the capacity to track food items across intricate supply chains, according to a case study by IBM Food Trust. Restaurants were able to reduce the danger of contamination and ensure the safety and quality of their ingredients by using the technology to track their

origin in real-time. The experiments at Walmart and Carrefour, which used blockchain technology in conjunction with data analysis tools driven by artificial intelligence, had similar success in lowering the occurrence of foodborne illnesses by tracking food goods from farm to table [16].

Artificial intelligence (AI) and blockchain technology work together to improve food safety and traceability. The massive amounts of data generated by blockchain systems present an opportunity for artificial intelligence to sift through and spot trends, outliers, and threats. Artificial intelligence systems can detect potential transit spoilage by, for instance, alerting users to anomalies in blockchain-stored temperature recordings. In addition to improving blockchains static records, AI's predictive skills allow for real-time decision-making and automatic actions in the event of a food safety standard breach [17].

A growing number of eateries and food manufacturers are investigating how artificial intelligence (AI) and blockchain technology may work together to make food more traceable and safer. To ensure they are in line with food safety rules, fast food restaurants like Chipotle and McDonald's have tested blockchain systems that can track the provenance of their cattle and fresh fruit. Through the provision of clear information on the route of their food from farm to fork, these endeavours not only increase food safety but also boost customer trust.

4 Methodology

Initially, sensors send data to an AI system for analysis while tracking the temperature, humidity, and other aspects of food storage in real time. By comparing the current state of affairs with past data and regulatory criteria, artificial intelligence systems anticipate possible hazards and highlight any gaps. To guarantee data integrity and traceability, all sensor data, AI analysis, and remedial actions are documented on a blockchain ledger. Stakeholders with transparent access to safety records include suppliers, regulatory agencies, and restaurant management thanks to this unchangeable ledger. Third, the use of smart contracts automates the verification of compliance, setting off alarms or remedial measures in the event that safety procedure violations are found. In conclusion, regular AI-powered audits utilizing blockchain data guarantee sustained security by forecasting potential threats and validating conformance patterns. The integration of blockchain transparency and AI's predictive powers fortifies food safety regulations while bolstering customer confidence.

Figure 3 shows how sensors, AI, blockchain, and smart contracts are integrated for food safety. To effectively demonstrate the process, a linked flow is used to highlight each stage, from gathering data to initiating corrective action.

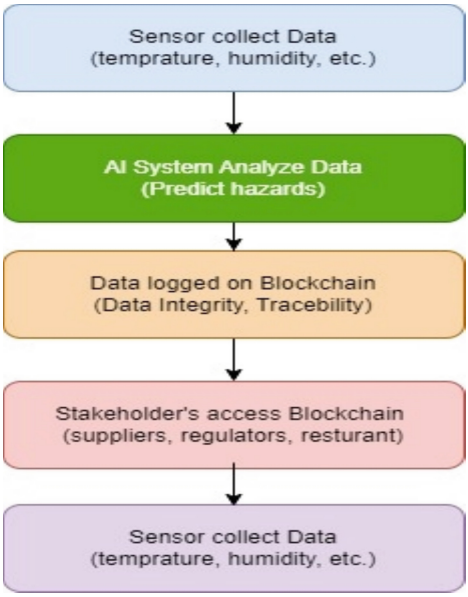


Fig. 3. Block diagram of System

4.1 Applications

- Data Collection

For food safety, there are several data sources that may be used, including internet databases, sensor data, genomics data, and social media data. But, problems arise from data variety, such as the difficulty in identifying pertinent information within a data source and its relationships to other data sources. Examples of food safety databases like GEMS/food, JECFA, RASFF, FDA Alert, FOSCOLLAB [2].

- Data Storage

To make data storage a reality, general data management methods like PostgreSQL, Oracle, and MySQL are used. Unfortunately, the SQL constraint on these systems' adaptability and durability makes them unsuitable for dealing with massive data. Therefore, open-source, horizontally scalable, next-generation databases are the target of NoSQL system development. Example NoSQL databases include MongoDB, Cassandra, and HBase.

- Data Analysis

Operations involving data analysis follow data storage and transfer to the processing unit. The Table 1 displays a compilation of the analytical tools that are most commonly utilized in ML and deep learning [18]. PYTHON is the basic programming language for data science and AI It integrates a broad scope of advanced ML algorithms for supervised and unsupervised problems [19].

Table 1. Examples of Data Storage, Processing, transferring and visualization [2].

Type	Technology	Tool	Resource
Data Storage	SQL	MySQL	http://www.mysql.com/
		Post-greSQL	http://www.postgresql.org/
		Oracle	http://www.oracle.com/
Data trans-ferring	Data trans-ferring	Aspera	http://asperasoft.com/
		Talend	https://www.talend.com/re-source/big-data-transfer.html
		Elas-ticsearch	https://www.elastic.co/
Data analysis	Platforms	Python	https://www.python.org/
		Scikit-learn	https://scikit-learn.org/stable/
		PyTorch	https://pytorch.org
		Tensor-Flow	https://www.tensorflow.org/
Data visuali-zation	Visualization	R	http://cran.r-project.org/
		Cisco	http://circos.ca/
		Fusion-Charts	http://www.fusioncharts.com/
		Gephi	https://gephi.github.io/
		GraphViz	http://www.graphviz.or/
		IBMMany Eyes	http://www.ibm.com/software/ana-lytics/many-eyes/

Table 1 explains the Professionals working with huge datasets may get a thorough overview of the many technologies, techniques, and resources utilized in various aspects of data management and processing in the table above. Relational databases such as PostgreSQL, Oracle, and MySQL provide strong data storage options for organizing structured data. Each database has a unique set of tools and a specialized platform that can be accessed via its website. Large datasets may be moved quickly and securely with the help of programs like Aspera and Talend, while Elasticsearch offers a scalable search engine for distributed systems. For complicated machine learning tasks in data analysis,

computer languages and libraries like Python, Scikit-learn, PyTorch, and TensorFlow—all of which have vibrant communities and copious documentation—are indispensable. In conclusion, a variety of platforms are available for generating visual representations of data in the field of data visualization, ranging from simple charts and graphs to intricate, interactive visualizations that promote deeper understanding and improved data communication. These tools include R, Cisco, FusionCharts, Gephi, GraphViz, and IBM Many Eyes. Together, these technologies provide an environment that facilitates all phases of the data lifecycle, from storage to analysis and display.

Human specialists often specify the characteristics utilized for food quality evaluation, which could lead to inaccurate categorization findings. When it comes to safeguarding food safety, deep learning may be preferable than employing food features individually since it can automatically learn food properties from photographs.

5 Result and Discussion

The following table provides an overview of the outcomes from every phase of the food safety system:

Table 2 shows how the system works and the advantages that are realized by outlining each phase of the process and the related result.

- **Enhanced Food Safety:** Predictive analytics and ongoing monitoring improve food storage's general safety.
- **Accountability and Transparency:** Because blockchain maintains a reliable record, stakeholders may watch compliance with confidence.
- **Automation:** By eliminating the need for human interaction and guaranteeing prompt response, smart contracts automate compliance checks.
- **Consumer Confidence:** AI and blockchain together promote trust by guaranteeing that the food that consumers eat is handled and kept securely.

Standards for food safety are strengthened by these findings, which support the integrated system's resilience and effectiveness.

6 Challenges and Futures

The Fig. 4 mentioned below the challenges of the applications of different Technologies.

Of utmost importance for data analysis is the data acquisition or collection procedure, which is responsible for the data quality. When evaluating the safety of food, AI is used for data processing and analysis. But AI model training requires a mountain of labeled data, which implies human specialists are still needed to anomalize the data. Computer resources are needed for training AI-based models, particularly deep learning-based models, because of data capacity and AI models.

Additionally, AI models frequently suffer from generalization issues, in which the trained models do not perform adequately in other contexts. One such cutting-edge technology is blockchain, which has just emerged for application in data security. Potential issues with fundamental infrastructure, technology, and data authenticity could be addressed.

Table 2. Result step by step

Stage	*Process*	*Results*
1. Sensors Collect Data	Sensors monitor humidity, temperature, and other aspects of storage.	Accurate data for additional analysis is ensured by real-time monitoring of crucial parameters.
2. AI Analyzes Data	AI contrasts recent data with records from the past and safety guidelines.	predicts possible problems, recognizes dangers before they become serious, and points up areas that need correction.
3. Blockchain Logging	AI analysis and data are stored on an unchangeable blockchain ledger.	guarantees the openness, traceability, and data integrity of food safety records.
4. Stakeholder Access	Restaurants, suppliers, and authorities may all access blockchain data.	Transparent access to safety records makes compliance monitoring and verification simple.
5. Smart Contracts	automated smart contract-based compliance checking.	if safety requirements are broken, sets off alarms or initiates corrective action.
6. Continuous Audits	routine audits driven by AI and supported by blockchain data.	maintains safety throughout time by anticipating hazards and confirming that safety procedures are being followed.
Overall System Outcome	Combining blockchain, AI, sensors, and smart contracts.	bolsters food safety regulations, boosts client assurance, and mechanizes compliance procedures.

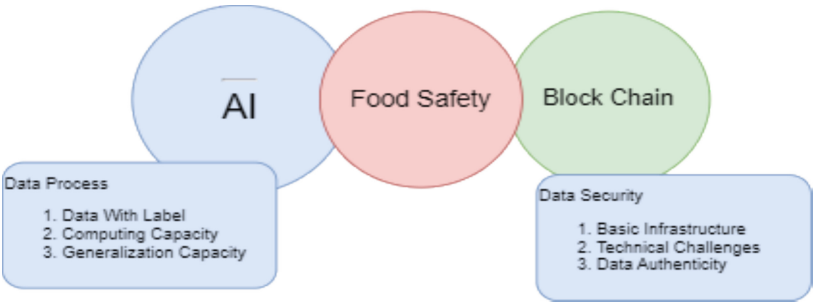


Fig. 4. Challenges of Artificial Intelligence and Blockchain for Food Safety

7 Conclusion

Since several cutting-edge technologies have been employed to guarantee food safety, food safety is an important subject in the fields of electrical engineering, computer science. This article explores the potential uses of artificial intelligence and blockchain in relation to food safety applications. We used a data-driven, holistic approach influenced by the food supply chain to provide an overview of these technologies. We looked at data gathering, storing, transmitting, analyzing, visualizing, and security through the lens of AI, and blockchain. Researchers in the subject of Food safety and information can benefit from this comprehensive review by getting a broader perspective.

References

1. Arkeman, Y., Hidayah, N.J., Suharso, A., Adhzima, F., Kusuma, T.: Implementation of artificial intelligence and blockchain in agricultural supply chain management. J. Int. Soc. Southeast Asian Agric. Sci. **29**(1), 135–149 (2023)
2. Zhou, Q., Zhang, H., Wang, S.: Artificial intelligence, big data, and blockchain in food safety. Int. J. Food Eng. **18**(1), 1–14 (2022). https://doi.org/10.1515/ijfe-2021-0299
3. Bandhu, K.C., Litoriya, R., Lowanshi, P., Jindal, M., Chouhan, L., Jain, S.: Making drug supply chain secure traceable and efficient: a Blockchain and smart contract based implementation. Multimed. Tools Appl. **82**(15), 23541–23568 (2023). https://doi.org/10.1007/s11042-022-14238-4
4. Tian, Y., Lu, Z., Adriaens, P., Minchin, R.E., Caithness, A., Woo, J.: Finance infrastructure through blockchain-based tokenization. Front. Eng. Manag. **7**(4), 485–499 (2020). https://doi.org/10.1007/s42524-020-0140-2
5. Litoriya, R., Arora, A., Bajaj, R., Gulati, A.: Adoption of blockchain technology in the indian business market: obstacles and opportunities. EAI/Springer Innov. Commun. Comput. 211–236 (2022). https://doi.org/10.1007/978-3-030-89546-4_11
6. Yele, S., Litoriya, R.: Blockchain-based secure dining: Enhancing safety, transparency, and traceability in food consumption environment. Blockchain Res. Appl. 100187 (2024). https://doi.org/10.1016/j.bcra.2023.100187
7. Yu, B., Zhan, P., Lei, M., Zhou, F., Wang, P.: Food quality monitoring system based on smart contracts and evaluation models. IEEE Access **8**, 12479–12490 (2020). https://doi.org/10.1109/ACCESS.2020.2966020
8. Tsoukas, V., Gkogkidis, A., Kampa, A., Spathoulas, G., Kakarountas, A.: Enhancing food supply chain security through the use of blockchain and TinyML. Inf. **13**(5) (2022). https://doi.org/10.3390/info13050213
9. Chen, Y.M., Chen, T.Y., Li, J.S.: A machine learning-based anomaly detection method and blockchain-based secure protection technology in collaborative food supply chain. Int. J. e-Collabor. **19**(1), 1–24 (2023). https://doi.org/10.4018/IJeC.315789
10. Gómez, J.M., Fares, O.H., Mohan, M., Lee, S.H.: Blockchain in the food industry: integrating machine learning in a systematic literature review. J. Int. Technol. Inf. Manag. **32**(1), 32–58 (2023). https://doi.org/10.58729/1941-6679.1569
11. Raparthi, B.M.: Blockchain-based supply chain management using machine learning : analyzing decentralized traceability and transparency solutions for optimized supply chain operations, vol. 1, no. 2, pp. 1–9
12. Jiang, Y., Lei, Y.: Implementation of trusted traceability query using blockchain and deep reinforcement learning in resource management. Comput. Intell. Neurosci. **2**, 2022 (2022). https://doi.org/10.1155/2022/6559517

13. Chen, H., Chen, Z., Lin, F., Zhuang, P.: Effective management for blockchain-based agri-food supply chains using deep reinforcement learning. IEEE Access **9**, 36008–36018 (2021). https://doi.org/10.1109/ACCESS.2021.3062410
14. Nandakumar, K., Sumathi, P., Sanjaykumar, S., Radhakrishnan, P., Vasanthi, N., Sandhiya, S.: Food safety transparency and traceability system based on ML enabled blockchain using design thinking. J. Posit. Sch. Psychol. **2022**(4), 5414–5424 (2022). http://journalppw.com
15. Ellahi, R.M., Wood, L.C., Bekhit, A.E.D.A.: Blockchain-based frameworks for food traceability: a systematic review. Foods **12**(16) (2023). https://doi.org/10.3390/foods12163026
16. Kamath, R.: Food traceability on blockchain: walmart's pork and mango pilots with IBM. J. Br. Blockchain Assoc. **1**(1), 1–12 (2018). https://doi.org/10.31585/jbba-1-1-(10)2018
17. Casino, F., Dasaklis, T.K., Patsakis, C.: A systematic literature review of blockchain-based applications: current status, classification and open issues. Telemat. Inform. **36**, 55–81 (2019). https://doi.org/10.1016/j.tele.2018.11.006
18. Shukla, S.S., Shukla, A.: On scientific computing. Intell. Instruments Comput. Appl. Lab. **3**(4), 19–20 (1985)
19. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems, NeurIPS, no. NeurIPS, pp. 8026–8037 (2019)



PIRNet: Two-Step Deep Neural Network for Segmentation of Brain MRI with Efficient Loss Functions

Ashis Datta(✉), Kunal Purkayastha, Palash Ghosal, and Rustom Ali Ahmed

Sikkim Manipal Institute of Technology, Sikkim Manipal University, Majitar, Sikkim 737136, India
{ashis.d,palash.g,rustam.a}@smit.smu.edu.in

Abstract. Segmentation of brain tissues from MRI scans is a critical first step in diagnosing brain-related issues. Traditionally, this task was performed manually by radiologists, which was time-consuming and prone to errors. Recently, machine learning techniques, especially deep learning, have been explored to automate this process with varying degrees of success. In this research, we propose the Pyramidal Inception Residual Network (PIRNet), a state-of-the-art model that combines Pyramid Pooling, inception blocks, and residual connections for improved brain MRI segmentation. Our model segments key tissues including cerebrospinal fluid (CSF), gray matter (GM), white matter (WM), and background (BG). We evaluate PIRNet on well-known public datasets: MRBrains13, iSeg17, and CANDI, and test it using various loss functions, such as Tversky loss, label-wise dice loss, categorical cross-entropy, and weighted categorical cross-entropy, to optimize segmentation accuracy.

The results show that PIRNet outperforms contemporary techniques with average dice scores of 94.55%, 91.28%, and 91.91% on the MRBrains13, iSeg17, and CANDI datasets, respectively. We also highlight the advantages of different loss functions, which enhance the model's robustness and architecture. Our method is versatile and can be applied to other segmentation tasks, such as breast cancer detection and liver lesion segmentation. PIRNet successfully tackles the complex challenges of volumetric brain tissue segmentation, offering substantial value for rapid diagnosis in clinical settings and advancing quantitative computer modeling in medical analysis.

Keywords: Convolutional neural network · Medical image segmentation · Brain MRI · Loss functions

1 Introduction

Magnetic Resonance Imaging (MRI), also known as MRI, provides a safe and non-invasive way to examine diverse tissues by creating high-quality brain images with various contrasts, surpassing other imaging techniques in both detail and

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 71–89, 2025.
https://doi.org/10.1007/978-3-031-86296-0_6

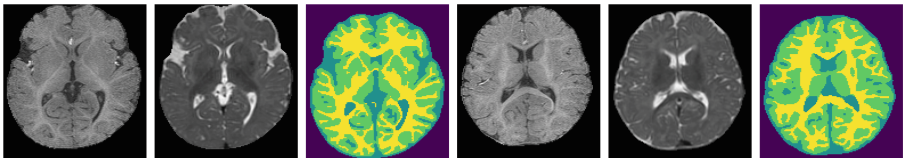


Fig. 1. Here is a demonstration showing consecutive MRI sections of two patients. The order of the images is as follows: T1 MRI, T1 MRI with gadolinium contrast, T2 MRI, FLAIR, and Labelled Ground Truth, from left to right.

flexibility [27]. Precise brain segmentation into its basic cytoarchitectural tissues is crucial for clinicians and radiologists in analyzing the brain quantitatively and qualitatively. This division is crucial in the diagnosis and treatment of different high-risk neurological conditions like epilepsy, schizophrenia, Alzheimer's disease, and dementia. Furthermore, it allows for the evaluation of uneven growth rates and sizes in various areas of the brain. Accurate tissue segmentation is crucial for studying brain development and detecting and treating symptoms of neurodevelopmental and neuropsychiatric conditions like schizophrenia, autism, ADHD, and perinatal brain injuries [7]. The initial year after birth is seen as the most active stage of postnatal brain development for newborns, according to research by Holland et al. (2014). Hence, it is essential to study the fast growth of brain tissues like Gray Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF) in these initial phases. Many proven and successful methods are built on traditional techniques [18]. Hence, it is crucial to examine the swift development of brain tissues like Gray Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF) in the initial phases. classical methods have been developed and proven successful. [10], as well as deep learning algorithms [5,6], have been implemented with promising results. Acclaimed classical models include Atlas-based models and parametric or deformable models [20,43]. More recently, several deep learning methods have come to light with advancements in Graphics Processing Unit (GPU) and Memory usage, burgeoning deep learning technology, and the absence of human participation in extracting the features [26]. Because of a limited amount of resources in the past, attempts were made to segment the regions patchwise in either small slices or Slice by Slice [17]. Lately, 3D deep learning convolutional models for segmentation with patchwise volumes have been proposed to realize the potential of the GPU and the memory available nowadays.

In this paper, we introduce a model fitted to automatically segment the CSF, GM, and WM from the 3D MRI volumes. This study is driven towards developing a Deep Learning architecture and examining the most desirable loss function for the task of brain MRI segmentation. The ultimate framework of the proposed model, PIRNet, is a combination of well-known architectures, the Inception network [38], and the Residual network [16] with the U-Net [33] designed to overcome the drawbacks of one other. Furthermore, we have included the state-of-the-art method called "pyramid pooling," which flawlessly recovers volumetric

context information and flawlessly merges multi-scale contextual feature representations. An ablation analysis has been performed to examine the efficacy of our model. We explore the strengths and weaknesses of each of the architectures, along with their interfusion. This paper also demonstrates the usage of various loss functions and their corresponding strengths, which further enhances the robustness of the proposed algorithm and the architecture as a whole.

Section 3 provides a thorough explanation of every operation. Section 4.1 covered the output images, experimental findings, and their analysis. Section 4.2 reviews the efficiency of the suggested method, and Sect. 4.3 considers the algorithm's potential future applications to wrap up the paper.

2 Related Works

MRI provides a non-invasive and secure way of examining various tissues by generating high-quality brain images in multiple contrasts compared to other imaging modalities [27]. Figure 1 depicts different MRI modalities used for differential diagnosis of brain-related disorders. Segmentation of the brain into its basic cytoarchitectural tissues is crucial for clinicians and radiologists in quantitative and qualitative analysis of the brain. It aids in diagnosing and treating numerous high-risk neurological disorders such as MS, Ataxia, schizophrenia, AD, and Parkinson's Disease. Additionally, it estimates the asymmetrical growth rate and size of several brain regions. Accurate tissue segmentation could be key in analyzing brain development and identifying and rectifying the signs of neurodevelopmental and neuropsychiatric diseases such as schizophrenia, autism, attention-deficit or perinatal brain injury, etc. [7] in advance for human brains. Therefore, studying the rapid growth of brain tissues such as GM, WM, and CSF in the initial stages becomes imperative. Many popular and effective approaches based on classical [10] and deep learning algorithms [5,6] have been implemented with promising results. Acclaimed classical models include Atlas-based models and parametric or deformable models [20,43]. More recently, several deep learning methods have come to light with advancements in Graphics Processing Unit (GPU) and Memory usage, burgeoning deep learning technology, and the absence of human participation in extracting the features [26]. Because of limited resources in the past, attempts were made to segment the brain regions patchwise in either small slices or slice by slice [17]. Lately, 3D deep learning convolutional models for segmentation with patchwise volumes have been proposed to realize the potential of the GPU and the memory available nowadays. This section briefly discusses some of the current methods for segmenting brain tissues from MR images. M.El-Melegy et.al. [11] presents a segmentation method on MRI brain Images in a nonparametric fashion. CSF, GM, WM, and Non-Brain Tissue are the four classifications into which the images are categorized. The paper claims their segmentation method shows increased performance even at higher degrees of noise and bias. Presented by Shanthi et al. [36] is a DICOM brain image segmentation system. It applies the thresholding method in conjunction with seed growth. According to the authors, their method produces encouraging

results and eliminates high-frequency speckles from the photographs. A Brain MRI segmentation system is proposed by Kumar et al. [22] using evolutionary computing approaches. According to the article, Robust Spatial Kernelled FCM with GA performs better than alternative FCM techniques. For CSF, WM, and GM, Anami et al. [2] have suggested an automated brain segmentation system. Modified FCM and level set-based approach form the foundation of the system. The authors come to the conclusion that segmenting using modified FCM produced satisfactory results. Huang et al. [19] presented a hybrid geometric deformable model for automatically segmenting brain tissues of 3D brain MRI images. The authors' approach can handle single and multiple MRI scan sequences. The model likewise offers consistent segmentation accuracy when segmenting the WM and GM tissues, with mean improvement percentages of 8.55 and 10.18. Ping Wang and HongLei Wang [42] provide a modified FCM segmentation module to segment the Brain MRI areas. According to the authors, the approach improves cluster performance for both the salt-pepper and Gaussian degraded images. Dongwei Gan et al. [12] have performed an experiment to get fine segmented results for the targeted area through dynamic routing algorithm without discarding any information from the 3D images. Guiyuan Zhu et al. [48] introduced an innovative fuzzy c-means (FCM) model that incorporates deep learning to enhance accuracy. They substituted Euclidean distance with a measurement function based on the skewed normal distribution to refine the segmented image. Farheen Ramzan et al. [32] developed a system for segmenting multiple brain regions utilizing 3D convolutional neural networks. Their method incorporates residual learning and dilated convolution operations to precisely map MRI volumes to voxel-level brain segments. The study aims to categorize up to nine brain areas, including cerebrospinal fluid, white matter, gray matter, and their subdivisions. In order to tackle the task of dividing brain MR images into two clusters, GM and WM, respectively, Agarwal et al. [1] proposed a bias field correction technique combined with the FCM segmentation. To segment the same regions, a level-set segmentation module was added to complete the process. According to the authors, the module's accuracy is better than that of the other methods. To address the segmentation and identification of the various brain MRI regions, Ortiz et al. [30] combine hierarchical SOM with probability clustering. The average metric values reported in the paper are 0.32 for WM, 0.75 for GM, and 0.69 for CSF. Valdes-Cristerna et al. [39] presented a hybrid model to segment multispectral MRI of a brain using a mix of a spline ACM and an RBF network. When evaluated using Tanimoto indices, the multispectral segmentation method performs better. Balafar et al. [3] had proposed a brain MRI segmentation model that was built upon a combination of GA and FCM. The center of the clusters discovered GA is initialized using the proposed model. The function is minimized in order to get there. In [25], a method for segmenting brain tissues using a patch-wise 3D U-net is presented. This network's encoding and decoding layers are modeled after the traditional U-net. For training, the network further uses overlapped 3D patches ($8 \times 24 \times 24$). Using transition layers with convolution operations between the ED layers differs from the traditional

U-Net architecture. This is done to highlight the significance of the feature maps that are incorporated into the decoding layers. A 2D (input patch size 13×13 pixels) patch-wise CNN technique was demonstrated by Zhang et al. [44] to segment WM, GM, and CSF from infants' multimodal (i.e., T1, T2, and fractional anisotropy) MR images. The study demonstrated that, in comparison to earlier techniques, the CNN strategy performs better. Additionally, they demonstrate improved performance compared to traditional machine learning techniques that include SVM and RF classifiers. Semantically-wise FCNs were constructed by Nie et al. [29] to segment baby brain pictures from the same dataset that Zhang et al. [44] utilized in their investigation. They achieved superior outcomes in contrast to [44]. For CSF, GM, and WM, the suggested approach produced an overall DSC of 85.5%, 87.3%, and 88.7%, respectively.

3 Dataset Description and Pre-processing

3.1 Dataset I: MRBrains

For the MICCAI grand competition for MR Brains segmentation, the MRBrains dataset is open to the public (https://mrbrains18.isi.uu.nl/). It consists of 30 patients' fully annotated T1-weighted, T1-weighted inversion recovery, and T2-FLAIR, bias-field corrected scans, all with the same alignment and voxel sizes of $0.958mm \times 0.958mm \times 3.0mm$. individuals with dementia, Alzheimer's disease, diabetes, cardiovascular risk, and varied degrees of atrophy and white matter lesions are among the individuals in this group who are older than 50. A 3T scanner at the UMC Utrecht was used to capture these pictures. There were only 7 patients in the training set.

3.2 Dataset II: ISeg

Ten newborns' pre-processed T1 and T2 weighted MRIs with manually segmented labels for CSF, WM, and GM were utilized in the MICCAI grand challenge dataset, which is accessible for 6-months neonatal brain MRI segmentation (iSeg-2017) [40]. With regard to the training data, the average ratio of (23278: 1376: 872: 641) for the BG, CSF, GM, and WM classes, respectively, roughly represents the number of voxels in each class label, which changes.

3.3 Dataset III: CANDI

The UMass Medical School Child and Adolescent Neuro Development Initiative (CANDI) provides the CANDI dataset [21]. It includes 103 patients' T1-weighted MRI brain scans from four diagnosis groups: Bipolar Disorder with Psychosis (BPD with Psy), Bipolar Disorder without Psychosis (BPD without Psy), Schizophrenia Spectrum (SS), and Healthy Controls (HC). It includes both linear and non-linear MNI152 2 mm skull-stripped scans; only the non-linear scans were utilized in this instance. With voxel spacing of 0.9 mm $\times$ 0.9 mm $\times$ 1.5 mm, all volumes had a dimension of $91 \times 109 \times 91$. For the BG, CSF, GM, and WM classes, the average ratio of (38318: 7367: 3440: 123) substantially represents the voxel distribution for each label (Table 1).

Table 1. Training, testing, and validation data split before and after augmentation.

Dataset	No of patients	Patches before aug			Patches after aug		
		Training	Testing	Validation	Training	Testing	Validation
MRBrains13	7	113	25	37	675	25	225
iSeg 2018	10	243	36	81	1215	36	405
CANDi Share	107	4678	252	1559	4678	252	1559

4 Methodology

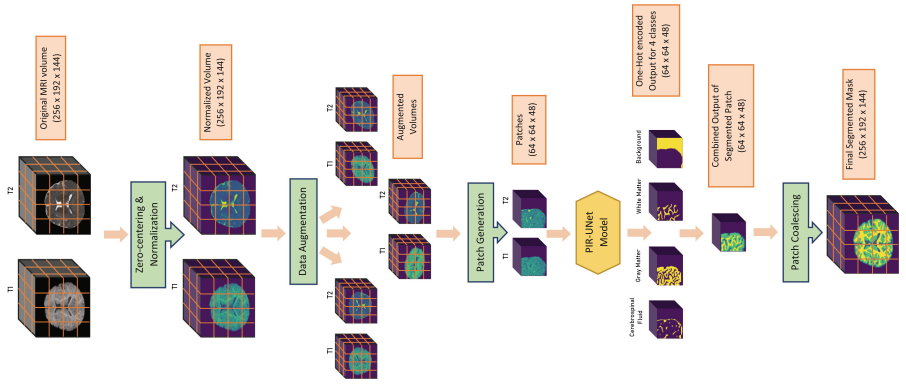


Fig. 2. A demonstrative flow diagram of the methodology.

4.1 Architecture

The architecture we have used to train all networks is 3D U-Net as proposed in [33] but with fewer learning parameters. The reduction was possible by removing a few convolution layers in each downsampling and upsampling layer. Two sets of inception layers were added to increase the efficiency of the convolution layers on the downsampling side. In a typical CNN layer, a choice is to be made to either have a stack of 3×3 filters, a stack of 5×5 filters, or a max pooling layer. The inception module in [38] suggests the use of all of them and lets the model decide by learning the appropriate weights given to those layers. Therefore, by doing this, the model gets an extra parameter to fit to the data as it deems fit. Another modification in the conventional 3D U-net is the usage of residual blocks in place of direct skip connections. Besides convolutions, a 3D CNN typically performs a down-sampling of feature maps across the network via pooling operations to fit the 3D data into memory and to increase the receptive field. The down-sampling strategy can lead to a loss of resolution in the segmentation,

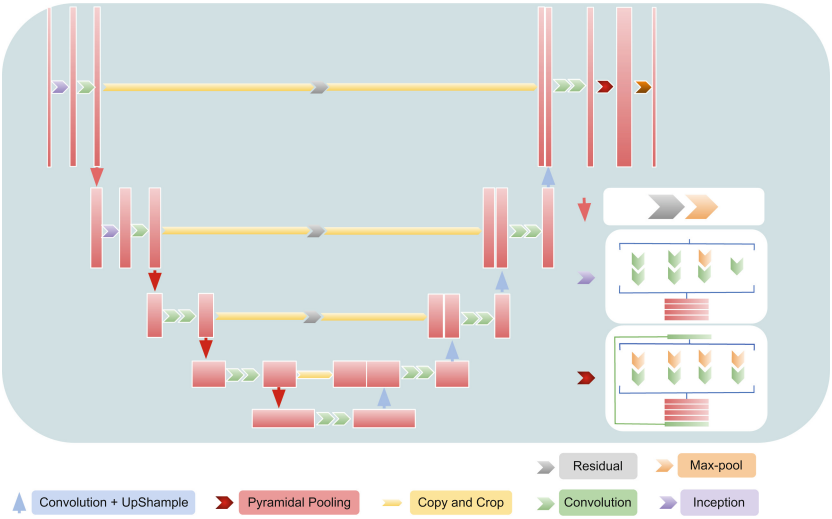


Fig. 3. An illustrative overview of the Inception- and Residual-based U-net architecture. Every dense block has five convolutional layers. With a growth rate of 16, each block produces 80 feature maps as its output.

which has been overcome in [24] by concatenating de-convolution (or transpose convolution) layers and a residual block from the corresponding down-sampling layer to recover the input image resolution in up-sampling. These blocks are used in place of identity unreferenced mapping, i.e., old skip-connection of U-net, to rescale the weights of vanishing down-sampling skip-connection layers in a more refined way and solve the issue of diminishing feature reuse in deeper U-net architecture. In particular, the shallower layers learn local features, while the deeper layers deal with the global features of the data. Experiments have also been done on using Pyramid Pooling (PP) by introducing them at the end of the upsampling part, improving the results. The comparison is given as a table in a later section. This concept was first introduced by [15] to perform spatial information aggregation of images at a deeper stage of the network between the last convolution layer and the fully connected layer, thus removing the fixed-size constraint of CNNs. Spatial PP, also known as spatial pyramid matching [14,23], also helps to combine the local features with the feature maps obtained at finer and coarser levels. As described [41], 3D pooling and convolution operations are used to form this module of filter sizes $1 \times 1 \times 1$, $4 \times 4 \times 4$ and $8 \times 8 \times 8$, each of which is then followed by a $1 \times 1 \times 1$ convolution layer. Due to this, the feature resolutions are reduced, and thus, to rescale them to the original feature map size, upsampling layers are used with the same filter size as the corresponding pooling layers. Finally, all the pyramid levels and their input are concatenated to get the aggregated global information. The detailed diagram of the proposed architecture, along with inception, residual, and PP module used in this paper, are given in Fig. 3. In the final layer after the up-sampling path, convolution

with 4 filters and a softmax activation layer was used to get the final predicted segmentation of four classes (CSF, GM, WM, and BG) (Fig. 2).

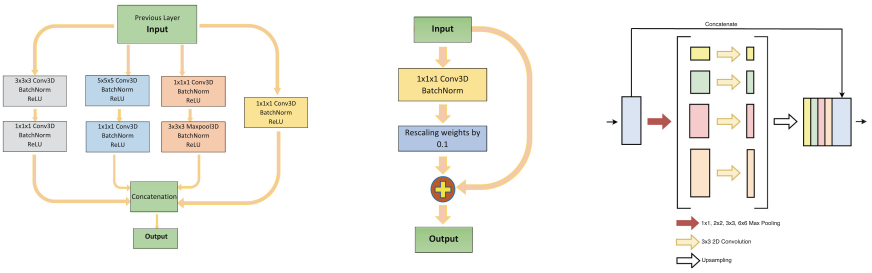
4.2 Training Strategy

After obtaining the pre-processing technique, we split our final dataset into three parts, i.e., training set, validation set, and testing set for training validation and testing of the proposed PIRNet Model. To make our model smaller and easily transferable, we train our model on patch data with input patch size 64×64 to utilize a smaller GPU. Adam is utilized as an optimizer for training the suggested design, starting at 5×10^{-3} for training. The technique we employed, as described in [8], involved reducing the learning rate by a factor of 0.5 in cases where the validation loss did not reduce over the course of five successive entries. It is observed that using cross-entropy loss and then fine-tuning using DSC loss obtains exceptionally high results as compared to only cross entropy loss, which was used before as a default.

$$L = -e^{\gamma} log\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right) + (1 - e^{\gamma}) \sum_j^C (1 - \frac{2\sum_i^N p_{ij} g_{ij} + \epsilon}{\sum_i^N p_{ij}^2 + \sum_i^N g_{ij}^2 + \epsilon}) \quad (1)$$

where s denotes the vector of output scores of one class of dimensionality C and s_p represents the class probability of simply the positive label and the number of classes. p_i and g_i in Eq. 1 are the voxel values of predicted class probabilities and the reference foreground mask, respectively, and N is the total number of voxels. The term ϵ is added as a smoothing factor to avoid any divide-by-zero error. This combination of loss functions is used for training during the MRBrains datasets. We trained our model to 200 epochs using appropriate early stopping conditions.

Post Processing and Testing. Since the model is trained on 64×64 patch data. Therefore, it will also take input data of shape 64×64 for result prediction on test data, and it will give predicted results in the shape of 64×64. The proposed PIRNet model has three outputs for GM, WM, and CSF with activation function sigmoid that will generate two labels: 0: background and 1: tissues(GM, WM, and CSF) in the predicted result. Therefore, we need to combine the results in a single patch by assigning 0, 1, 2, and 3 for background, CSF, GM, and WM, respectively. Then, Patch Coalescing on the 2D patches is performed to get the full segmented labeled volume. To check the robustness of the model further, we tested it using the test data. Refer to Table 2 for the evaluated results on each of the datasets, which are further discussed in the latter sections. Since the model is trained using a 5-fold cross-validation strategy. Therefore to make a stable prediction, a voting mechanism is used as well as shown in Fig. 5



(a) (b) (c)

Fig. 4. Detailed structure of (a) Inception Block (b) Residual Block (c) Pyramid Pooling Block.

Table 2. Quantitative results; DSC, ASD, and MHD of the final model on MRBrains, ISeg2017, and CANDI datasets for all 3 major types of tissues: CSF, GM, and WM. All the results shown in the table are the average values of the 3D-predictable results for each patient.

Dataset	Dice Coefficient				ASD				MHD			
	CSF	GM	WM	Average	CSF	GM	WM	Average	CSF	GM	WM	Average
Training Set												
ISeg2017	0.9566	0.9402	0.9116	0.9361	0.9675	0.9231	0.8067	0.8991	0.7033	0.9371	0.8567	0.8324
MRBrains	0.9930	0.9300	0.9405	0.9545	0.9644	0.9862	0.7093	0.8866	0.7248	0.8123	0.7891	0.7754
CANDI	0.7780	0.9113	0.8567	0.8487	0.8938	0.6320	0.8371	0.7876	0.7497	0.8081	0.9321	0.8300
Validation Set												
ISeg2017	0.9483	0.9265	0.9042	0.9263	0.9521	0.9203	0.8120	0.8948	0.7150	0.9276	0.8390	0.8272
MRBrains	0.9910	0.9202	0.9387	0.9500	0.9610	0.9801	0.7154	0.8855	0.7334	0.8203	0.7930	0.7822
CANDI	0.7680	0.9011	0.8454	0.8382	0.8830	0.6210	0.8260	0.7767	0.7590	0.7970	0.9221	0.8260
Test Set												
ISeg2017	0.9401	0.9182	0.8954	0.9179	0.9470	0.9115	0.8202	0.8929	0.7255	0.9183	0.8215	0.8218
MRBrains	0.9889	0.9101	0.9283	0.9424	0.9589	0.9741	0.7205	0.8845	0.7421	0.8283	0.8001	0.7902
CANDI	0.7589	0.8910	0.8343	0.8281	0.8739	0.6100	0.8150	0.7663	0.7691	0.7860	0.9123	0.8225

4.3 Evaluation Metrics

Three evaluation metrics-the Dice coefficient (DC) [35], the Modified Hausdorff distance (MHD) [13], and the Average Surface Distance (ASD) [34]-are used to assess the model performance for iSeg-2017 and IBSR18. These metrics are calculated separately for each GM, WM, and CSF tissue. Equations 2, 3, and 4 define DC, ASD, and MHD for the anticipated result $\hat{y}$ and the matching ground truth y. The Euclidean distance between two points a and b is represented by $d(a, b) = \|a - b\|$.

$$DC(y, \hat{y}) = \frac{2 \sum y\hat{y}}{\sum y + \sum \hat{y}} \tag{2}$$

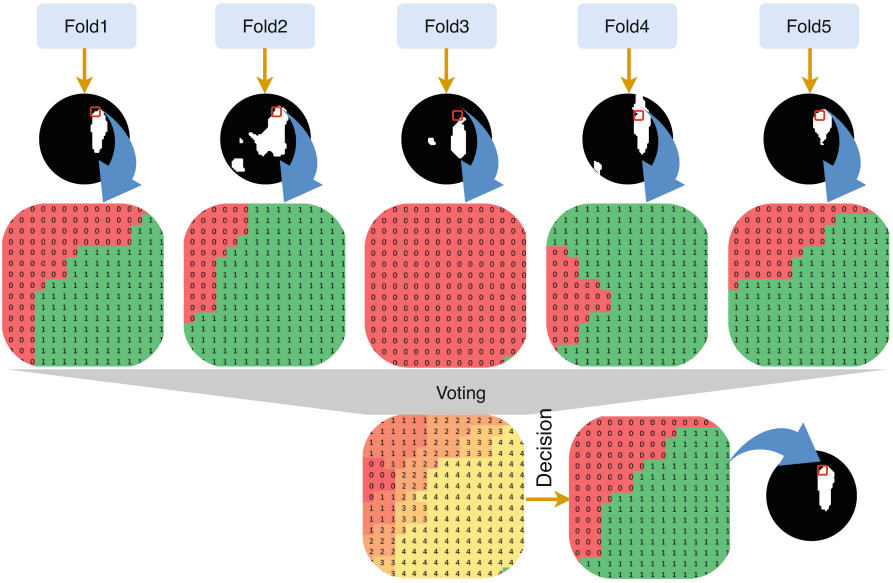


Fig. 5. voting mechanisum.

$$ASD(y,\hat{y}) = \frac{1}{2}\left(\frac{\sum_{a\in y} min_{b\in\hat{y}} d(a,b)}{\sum y} + \frac{\sum_{b\in\hat{y}} min_{a\in y} d(b,a)}{\sum \hat{y}}\right) \tag{3}$$

$$MHD(y,\hat{y}) = max\left(\frac{\sum_{a\in y} min_{b\in\hat{y}} d(a,b)}{\sum y}, \frac{\sum_{b\in\hat{y}} min_{a\in y} d(b,a)}{\sum \hat{y}}\right) \tag{4}$$

The degree of overlap between the segmentation findings and the ground truth is measured by the $0 \leq DC \leq 1$. While MHD matches two objects based on their edge point, ASD calculates the average distance between the surface points. ASD and MHD values that are lower indicate greater item matching.

5 Experimental Results and Comparisons

5.1 Experimental Setup

Using a Tesla K80 with 2496 CUDA cores, compute 3.7, 12 GB (11.439 GB Usable) GDDR5 VRAM on a machine with 24 GB RAM and a core-i7 7th generation @ 4.0 GHz processor, the entire configuration was trained in a Windows environment. Every network was built using the Python program Keras.

5.2 Experiments and Results (Ablation Analysis on Training, Testing and Validation Set for MRBrains Dataset)

Effectiveness of Residual Block. The universal approximation theorem declares that a single-layer feedforward network has the capability to represent any function, provided it has sufficient capacity. Nevertheless, these networks may be susceptible to overfitting because of their potentially extensive magnitude. Hence, it is commonly believed in research that network structures should be made deeper. Although deeper networks can improve accuracy by preventing overfitting, they also present difficulties due to the vanishing gradient problem, where the signal needed to adjust weights decreases in early layers. Adding layers without proper consideration can lead to a higher risk of training error due to the increased complexity and difficulty of optimization in a larger parameter space. By building the network using modules known as residual models, as seen in the Fig. 4b, residual networks [16] enable training of such deep networks. The term for this is deterioration problem Prior to Residual Networks, there were a number of approaches to address the vanishing gradient [38] issue, such as incorporating an additional loss as extra supervision in a middle layer. Nevertheless, none of these methods completely solve the problem. ResNet's "identity shortcut connection" concept tackles this issue by enabling stacked layers to learn residual mappings effectively. The results also underscore the importance of residual connections in enabling depth and enhancing the functionalities of deep neural networks.

Inception Block. Selecting the appropriate kernel size for the convolution process becomes difficult because of the wide variance in the information's position. Information dispersed more locally is favored with a smaller kernel, whereas information scattered more globally is better with a larger kernel. Over-fitting is a problem with very deep networks. Additionally, gradient adjustments are difficult to propagate throughout the network. Large convolution processes are computationally costly when stacked naively. If the module were made deeper, there would be an extreme reduction in dimensions and consequent information loss. The representational bottleneck was eliminated by widening, not deepening, the filter banks in the module. The module functions as a number of pooled convolution filters applied to the same input. After that, the outcomes are concatenated. This enables the model to utilize multi-level feature extraction. For example, it simultaneously extracts local (1×1) and universal (5×5) features. The network performs better when numerous features from several filters are used. Apart from that, the conception of architecture is superior to others due to another factor. Prior to creation, all architectures executed convolution jointly on the spatial and channel-wise domains. The inception block ignores the spatial dimensions and performs cross-channel correlations by executing the 1×1 convolution. Cross-spatial and cross-channel correlations are then obtained using the 3×3 and 5×5 filters. The findings of the aforementioned study show how the inception block might enhance network performance.

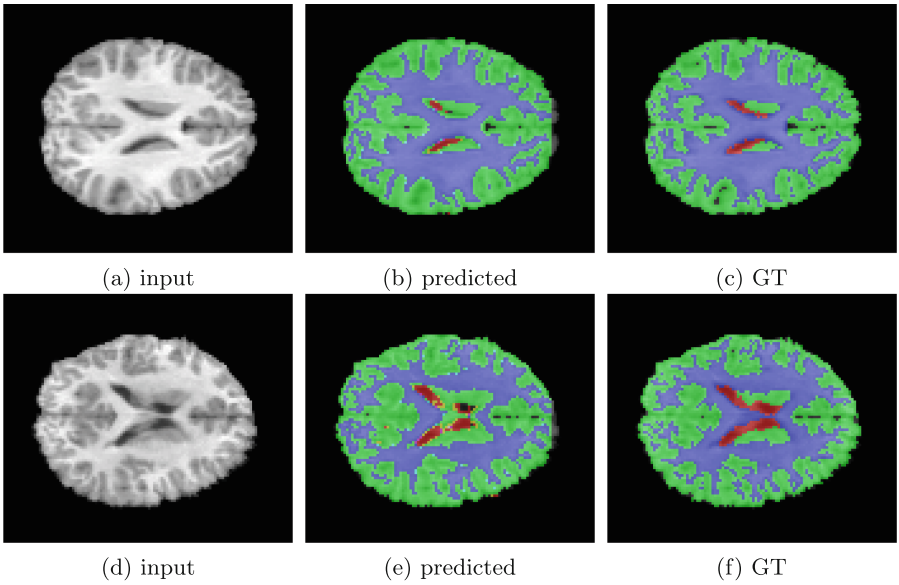
Pyramid Pooling. CNN was first exposed to pyramid pooling in SPP-Net [15]. Pyramid pooling is a technique that allows convolutional neural networks to be freed from their fixed size constraints while maintaining robustness against object deformations and loss of picture position information. Pyramid pooling produces feature maps at various levels in PSPNet [46]. These feature maps are combined by the pyramid pooling module to provide local and global context data. As illustrated in Fig. 3, our network's pyramid pooling module includes four pyramid levels. We perform 3D pooling operators to employ volumetric information instead of just spatial information. $1 \times 1 \times 1$, $4 \times 4 \times 4$, $8 \times 8 \times 8$, and $16 \times 16 \times 16$ are the bin sizes used in pyramid pooling. The feature map is divided into several sub-volumes by the various pyramid levels, which also obtain the pooled feature representation at various points. Every pyramid level has a $1 \times 1 \times 1$ convolution layer after it to get the total number of channels down to 1/4. The low-dimension feature is rescaled to the original feature map shape using an upsampling layer. The last global feature of pyramid pooling is the concatenation of many feature levels.

Table 3. An ablation study of our architecture based on Loss and Dice-coefficient

Model	Training		Validation	
	Loss	DSC (CSF/GM/WM)	Loss	DSC (CSF/GM/WM)
Model0: Baseline 3D U-Net	0.5587	0.8755/0.8829/0.9077	0.5049	0.8899/0.8961/0.9189
Model1: 3D U-Net	0.5721	0.8691/0.8816/0.9065	0.5122	0.8862/0.8951/0.9184
Model2: 3D U-Net + Residual	0.6284	0.8542/0.8704/0.9008	0.5786	0.8665/0.8823/0.9099
Model3: 3D U-Net + Inception	0.6358	0.8597/0.8693/0.8952	0.5891	0.8722/0.8806/0.9051
Model4: 3D U-Net + Inception + Residual	0.5140	0.8835/0.8950/0.9180	0.4594	0.8987/0.9079/0.9293
Model5: 3D U-Net + Pyramid Pooling (PP)	0.5140	0.8835/0.8950/0.9180	0.4594	0.8987/0.9079/0.9293
Model6: 3D U-Net + Inception + PP	0.5199	0.8825/0.8933/0.9164	0.4655	0.8974/0.9069/0.9278
Model7: 3D U-Net + Residual + PP	0.4928	0.8880/0.8992/0.9201	0.4451	0.9015/0.9110/0.9292
Model8: PIRNet (3D U-Net + Inception + Residual + PP)	0.4744	0.8918/0.9040/0.9257	0.4252	0.9057/0.9164/0.9359

We compare our network with and without the pyramid pooling module in order to verify the module's efficacy. Table 3 illustrates the improvement in average DSC on training and testing datasets that our network with the Pyramid Pooling Module (PIRNet) obtained compared to the network without the module, with improvements of 2.61% and 6.75%, respectively.

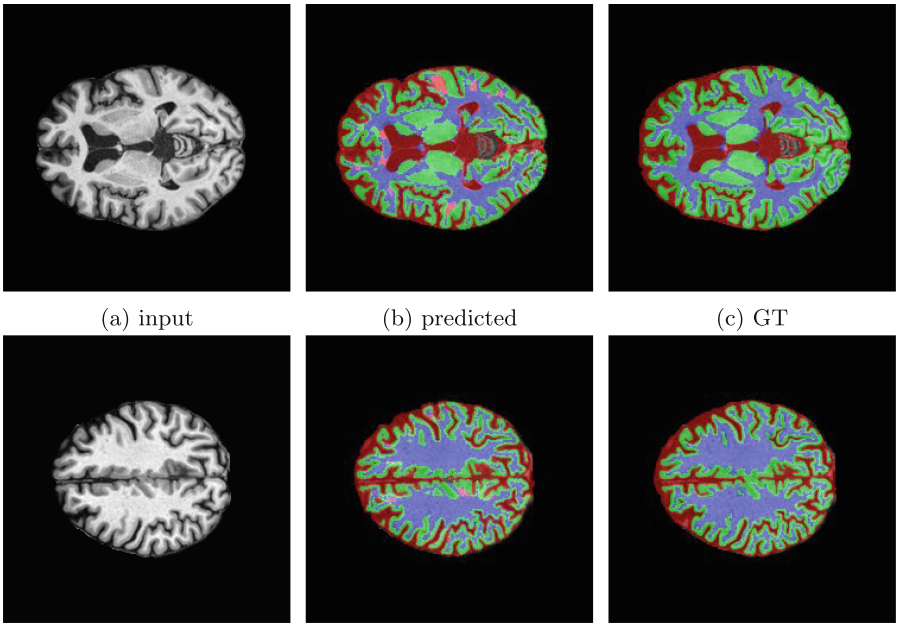
Cross-Entropy with Dice Loss. Cross-entropy loss at the pixel level is frequently employed in tasks related to segmenting images. This loss function assesses each pixel separately by comparing the predicted class probabilities (obtained from pixel vectors) with the target vector encoded in one-hot format. Calculating the average cross entropy loss for all pixels gives every pixel equal importance in the learning process. Yet, an imbalance in the distribution of classes within the image can create difficulties since the class that appears most frequently may have a disproportionate impact on the training process. The authors [37] address the issue of class imbalance in the dataset by addressing



(a) input (b) predicted (c) GT

(d) input (e) predicted (f) GT

Fig. 6. Segmented result on CANDI dataset, the first column represents the input, second and third represent its predicted and ground truth of CSF, GM, WM.



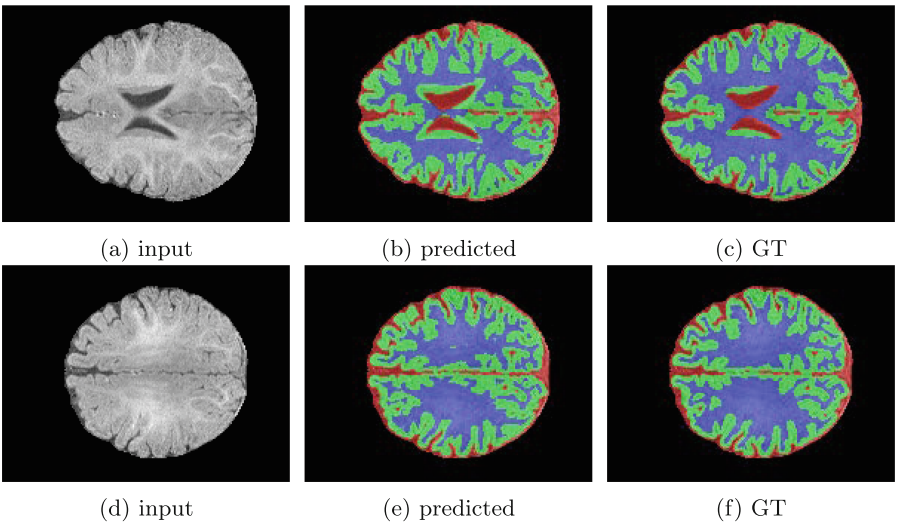
(a) input (b) predicted (c) GT

(d) input (e) predicted (f) GT

Fig. 7. Segmented result on MRBrains dataset, the first column represents the input, second and third represent its predicted and ground truth of CSF, GM, WM.

Table 4. Comparison of final mean-IOU and dice-coefficient on training of model using different loss functions

Loss Function	Training		Validation	
	Loss	DSC (CSF/GM/WM)	Loss	DSC (CSF/GM/WM)
Binary Cross Entropy	0.5721	0.9083/0.9100/0.9302	0.5122	0.9089/0.9110/0.9307
Dice Loss	0.6358	0.8675/0.8789/0.9011	0.5891	0.8672/0.8788/0.9016
BCE_Dice_loss	0.4744	0.9369/0.9400/0.9517	0.4252	0.9371/0.9402/0.9516
Tversky Loss	0.6284	0.8566/0.8554/0.8948	0.5786	0.8567/0.8558/0.8927
Poission Loss	0.5140	0.9221/0.9273/0.9449	0.4594	0.9223/0.9278/0.9451
Focal Loss	0.5199	0.8700/0.8781/0.9160	0.4655	0.8699/0.8786/0.9168
Tversky Focal Loss	0.4928	0.8514/0.8541/0.8988	0.4451	0.8514/0.8546/0.8987



(a) input (b) predicted (c) GT

(d) input (e) predicted (f) GT

Fig. 8. Segmented result on Iseg2017 dataset, the first column represents the input, second and third represent its predicted and ground truth of CSF, GM, WM.

the weighting of this loss for every output channel. In essence, the Dice coefficient measures how much two samples overlap. On a scale of 0 to 1, a Dice coefficient of 1 represents perfect and total overlap. This loss function is called the "soft Dice loss" because, rather than thresholding and putting the predicted probabilities into a binary mask, we use them directly. Regarding the neural network output, the quantity of activations in each mask independently is of interest to the denominator, whereas the shared activations between our forecast and the target mask are of interest to the numerator. This normalizes our loss based on the target mask's size, ensuring that the soft Dice loss learns from classes with a lower spatial representation in an image without difficulty. Dice loss was also created for the semantic segmentation of classes that were not balanced. Because

it incorporates the benefits of both class-level and voxel-level loss, the hybrid loss between dice loss and cross-entropy loss performs better than dice loss. A hybrid loss function that combines cross entropy and dice loss gives the optimal overall dice coefficient. Comparison of final mean-IOU and dice-coefficient on the training of model using different loss functions is presented in Table 4.

5.3 Quantitative and Qualitative Analysis of Results

The respective models were evaluated on the unseen testing data, which was separated from the actual dataset. Following are the results for each dataset. It can be observed that in the CANDI dataset, which can be understood from Fig. 6, results for CSF are not on par with the other modalities. The contribution of CSF voxels in the whole dataset, which is just 0.25%, could be the reason that our model weights were not adjusted to their voxel intensity even after balancing the dataset.

Table 5. The DSC values of the segmentation methods on brain MR images.

Dataset	Method	DSC		
		GM	WM	CSF
MRBrains13	Zhang et al., 2015 [45]	0.864	0.852	0.835
	Nie et al., 2016 [29]	0.887	0.873	0.855
	Nguyen et al., 2017 [28]	0.90	0.91	0.79
	F Ramzan et al., 2019 [32]	0.872	0.872	0.896
	G Zhu et al., 2024 [48]	68.75	72.53	79.35
	Proposed method	**91.01**	**92.83**	**98.89**
iSeg-2017	3D U-Net, 2016 [9]	89.83	90.55	94.39
	nnFormer, 2021 [47]	91.66	91.80	94.21
	VT-UNet, 2022 [31]	91.23	92.26	94.78
	Bhairnallykar et al., 2023 [4]	91.18	93.08	95.60
	3D-EffiViTCaps, 2024 [12]	93.59	**93.61**	95.61
	Proposed method	**94.02**	91.16	**95.66**

Table 5 presents the DSC scores for various segmentation methods applied to real brain MRI images across two datasets: MRBrains13 and iSeg-2017, categorized by tissue types: Gray Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF). For the MRBrains13 dataset, which can be understood from Fig. 7, significant contributions from studies such as Farheen Ramzan et al. (2019) and Guiyuan Zhu et al. (2024) are highlighted, with the proposed method achieving an impressive DSC of 91.01 for GM, 92.83 for WM, and 98.89 for CSF, reflecting a substantial improvement over earlier methods. In the iSeg-2017 dataset, which can be understood from Fig. 8, the results showcase established techniques like 3D U-Net and nnFormer, alongside the proposed method,

which further demonstrates effectiveness with DSC values of 94.02 for GM, 91.16 for WM, and 95.66 for CSF.

6 Conclusions

Research on brain tissue segmentation has been conducted recently. The work offers a straightforward framework for efficient brain tissue segmentation from MR images based on deep learning. The work makes numerous contributions. We undertake a quantitative analysis utilizing a wide range of performance criteria to show how well our methodology performs in comparison to some other cutting-edge brain MRI segmentation techniques. Determining which network architecture changes result in the biggest improvements will require an ablation study concerning many components and the loss function. Validation of the suggested strategy on a well-known publicly available dataset yields encouraging results. Every difficulty encountered during the volumetric segmentation of the brain tissues is resolved by this technique. It can be used for several segmentation issues, including the segmentation of liver lesions and the detection of breast cancer, among others. We think our method has a lot of promise in quantitative accurate computer modeling when fast interactions are needed for therapeutic applications.

References

1. Agarwal, P., Kumar, S., Singh, R., Agarwal, P., Bhattacharya, M.: A combination of bias-field corrected fuzzy c-means and level set approach for brain MRI image segmentation. In: 2015 Second International Conference on Soft Computing and Machine Intelligence (ISCMI), pp. 84–87. IEEE (2015)
2. Anami, B.S., Unki, P.H.: A combined fuzzy and level sets' based approach for brain MRI image segmentation. In: 2013 Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), pp. 1–4. IEEE (2013)
3. Balafar, M., Rahman Ramli, A., Saripan, M.I., Mashohor, S., Mahmud, R.: Medical image segmentation using fuzzy c-mean (FCM) and user specified data. J. Circ. Syst. Comput. **19**(01), 1–14 (2010)
4. Bhairnallykar, S.T., Narawade, V.: Segmentation of MR images using DN convolutional neural network. Int. J. Inf. Technol. **15**(8), 4565–4576 (2023)
5. de Brebisson, A., Montana, G.: Deep neural networks for anatomical brain segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 20–28 (2015)
6. Chen, H., Dou, Q., Yu, L., Qin, J., Heng, P.A.: VoxResNet: deep voxelwise residual networks for brain segmentation from 3d mr images. Neuroimage **170**, 446–455 (2018)
7. Choe, M.s., et al.: Regional infant brain development: an MRI-based morphometric analysis in 3 to 13 month olds. Cerebral Cortex **23**(9), 2100–2117 (2012)
8. Chollet, F., et al.: Keras (2015). https://keras.io

9. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D u-net: learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 424–432. Springer (2016). https://doi.org/10.1007/978-3-319-46723-8_49
10. Despotović, I., Goossens, B., Philips, W.: MRI segmentation of the human brain: challenges, methods, and applications. Comput. Math. Methods Med. **2015** (2015)
11. El-Melegy, M.T., Mokhtar, H.M.: Tumor segmentation in brain MRI using a fuzzy approach with class center priors. EURASIP J. Image Video Process. **2014**, 1–14 (2014)
12. Gan, D., Chang, M., Chen, J.: 3D-EffiviTCaps: 3D efficient vision transformer with capsule for medical image segmentation. arXiv preprint arXiv:2403.16350 (2024)
13. Gao, Y.: Efficiently comparing face images using a modified hausdorff distance. IEE Proc. Vis. Image Signal Process. **150**(6), 346–350 (2003)
14. Grauman, K., Darrell, T.: The pyramid match kernel: discriminative classification with sets of image features. In: Tenth IEEE International Conference on Computer Vision (ICCV 2005) Volume 1, vol. 2, pp. 1458–1465. IEEE (2005)
15. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2015)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
17. Hesamian, M.H., Jia, W., He, X., Kennedy, P.: Deep learning techniques for medical image segmentation: achievements and challenges. J. Digit. Imaging, 1–15 (2019). https://doi.org/10.1007/s10278-019-00227-x
18. Holland, D., et al.: Structural growth trajectories and rates of change in the first 3 months of infant brain development. JAMA Neurol. **71**(10), 1266–1274 (2014)
19. Huang, A., Abugharbieh, R., Tam, R.: A hybrid geometric-statistical deformable model for automated 3-D segmentation in brain MRI. IEEE Trans. Biomed. Eng. **56**(7), 1838–1848 (2009)
20. Išgum, I., et al.: Evaluation of automatic neonatal brain segmentation algorithms: the neobrains12 challenge. Med. Image Anal. **20**(1), 135–151 (2015)
21. Kennedy, D.N., Haselgrove, C., Hodge, S.M., Rane, P.S., Makris, N., Frazier, J.A.: Candishare: a resource for pediatric neuroimaging data (2012)
22. Kumar, S.A., Harish, B., Guru, D.: Segmenting MRI brain images using evolutionary computation technique. In: 2015 International Conference on Cognitive Computing and Information Processing (CCIP), pp. 1–6. IEEE (2015)
23. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), vol. 2, pp. 2169–2178. IEEE (2006)
24. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
25. Luna, M., Park, S.H.: 3D patchwise U-Net with transition layers for MR brain segmentation. In: International MICCAI Brainlesion Workshop, pp. 394–403. Springer (2018). https://doi.org/10.1007/978-3-030-11723-8_40
26. Lundervold, A.S., Lundervold, A.: An overview of deep learning in medical imaging focusing on MRI. Z. Med. Phys. **29**(2), 102–127 (2019)

27. McBrien, M., Winder, J., Smyth, L.: Anaesthesia for magnetic resonance imaging: a survey of current practice in the UK and Ireland. Anaesthesia **55**(8), 737–743 (2000)
28. Nguyen, D.M., Vu, H.T., Ung, H.Q., Nguyen, B.T.: 3D-brain segmentation using deep neural network and gaussian mixture model. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 815–824. IEEE (2017)
29. Nie, D., Wang, L., Gao, Y., Shen, D.: Fully convolutional networks for multi-modality isointense infant brain image segmentation. In: 2016 IEEE 13Th International Symposium on Biomedical Imaging (ISBI), pp. 1342–1345. IEEE (2016)
30. Ortiz, A., Gorriz, J., Ramírez, J., Salas-Gonzalez, D.: MR brain image segmentation by growing hierarchical SOM and probability clustering. Electron. Lett. **47**, 585–586 (2011)
31. Peiris, H., Hayat, M., Chen, Z., Egan, G., Harandi, M.: A robust volumetric transformer for accurate 3d tumor segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 162–172. Springer (2022). https://doi.org/10.1007/978-3-031-16443-9_16
32. Ramzan, F., Khan, M., Iqbal, S., Saba, T., Rehman, A.: Volumetric segmentation of brain regions from MRI scans using 3D convolutional neural networks. IEEE Access **8**, 103697–103709 (2020)
33. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
34. de Santos-Berbel, C., Essa, M., Sayed, T., Castro, M.: Reliability-based analysis of sight distance modelling for traffic safety. J. Adv. Transp. **2017**(1), 5612849 (2017)
35. Shamir, R.R., Duchin, Y., Kim, J., Sapiro, G., Harel, N.: Continuous dice coefficient: a method for evaluating probabilistic segmentations. arXiv preprint arXiv:1906.11031 (2019)
36. Shanthi, S., et al.: A novel method for pneumothorax diagnosis and segmentation using deep convolution neural network. In: 2023 International Conference on Computer Communication and Informatics (ICCCI), pp. 1–8. IEEE (2023)
37. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**, 1–1 (05 2016)
38. Szegedy, C., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp. 1–9 (2015)
39. Valdés-Cristerna, R., Medina-Bañuelos, V., Yáñez-Suárez, O.: Coupling of radial-basis network and active contour model for multispectral brain MRI segmentation. IEEE Trans. Biomed. Eng. **51**(3), 459–470 (2004)
40. Wang, L., et al.: Benchmark on automatic 6-month-old infant brain segmentation algorithms: the iSeg-2017 challenge. IEEE Trans. Med. Imaging **38**, 2219–2230 (2019)
41. Wang, L., Xie, C., Zeng, N.: RP-Net: a 3D convolutional neural network for brain segmentation from magnetic resonance imaging. IEEE Access **7**, 39670–39679 (2019)
42. Wang, P., Wang, H.: A modified FCM algorithm for MRI brain image segmentation. In: 2008 International Seminar on Future BioMedical Information Engineering, pp. 26–29. IEEE (2008)
43. Weisenfeld, N.I., Warfield, S.K.: Automatic segmentation of newborn brain MRI. Neuroimage **47**(2), 564–572 (2009)
44. Zhang, F., et al.: Deep learning based segmentation of brain tissue from diffusion MRI. Neuroimage **233**, 117934 (2021)

45. Zhang, W., et al.: Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. Neuroimage **108**, 214–224 (2015)
46. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2881–2890 (2017)
47. Zhou, H.Y., Guo, J., Zhang, Y., Yu, L., Wang, L., Yu, Y.: nnFormer: interleaved transformer for volumetric segmentation. arXiv preprint arXiv:2109.03201 (2021)
48. Zhu, G., Liao, S., Zhan, T., Chen, Y.: An improved fuzzy c-means method based on multivariate skew-normal distribution for brain MR image segmentation. KSII Trans. Internet Inf. Syst. **18**(8), 2082–2102 (2024)

Neural Networks Approach for Liver Cancer Classification

Vijaya Lakshmi Adluri(✉), Mohan Aditya Dondapati, Maadesh Darisi, John Paul Devapooja, and Kaushik Chintalapati

B V Raju Institute of Technology, Narsapur, Medak District, Telangana 502313, India
vijayaadlurik@gmail.com

Abstract. Due to its high death rate, liver cancer continues to pose serious threat to world health. Tumors growing on the liver can cause liver cancer. This liver malignancy has several different forms. Cancer of the liver is frequent yet severe. The most common cause of this liver cancer is chronic liver disease, such as cirrhosis from hepatitis B or hepatitis C. Long-term liver problems are associated with an increased risk of liver cancer, the most frequent type of liver cancer. Medications may include chemotherapy, hardening or heating the cancer cells, transplantation, and surgery. In this work, we employ neural networks to analyze data and identify important characteristics for the identification of liver cancer.

Keywords: Machine Learning · neural networks · deep learning · liver cancer

1 Introduction

Liver cancer is one of the most common threatening diseases. Liver cancer considered to be metastatic. Symptoms of Liver Cancer may begin as a lump on the liver, progressing to hepatic complications. Liver cancer or bile duct cancer, arises from the epithelial cells lining the bile ducts. Today, traditional methods of diagnosing such devastating diseases are compromised. It is possible that unusual cells will ultimately cause the victims to pass away. Another major factor contributing to its difficulty to treat is repetition. When diagnosing liver cancer, taking biological data into account can produce accurate results. Depending on the kind of dataset, the machine learning models include CNN, RNN, RF, SVM, and many more. The initial stage in diagnosing a typical case of liver cancer involves tracking the data's sequential changes.

2 Literature Survey

Based on a number of metrics and statistical tests, a novel staging model for hepatocellular carcinoma (HCC) tenthly shows greater discrimination power over the BCLC approach. Comparable homogeneity ratings between the proposed model and the BCLC method show that patient staging is fairly homogeneous. It is demonstrated that the CHF embedding has the strongest predictive value when compared to other patient representations, which helps to produce survival forecasts that are more accurate. The medical

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 90–98, 2025.
https://doi.org/10.1007/978-3-031-86296-0_7

industry is seeing a tremendous expansion in the use of big data and artificial intelligence, which presents opportunities for more thorough and efficient patient staging and classification. Patients gave their informed consent to participate in the study, which was approved by the Ethics Committee and supported by a number of research funding [1].

A two-step method was employed because the training data was uneven to distinguish between HCC, CC, and mCRC. The SSMH dataset's classification performance was considerably enhanced using a classifier trained on mixed datasets. All AUCs for the SSMH dataset were 1.000, indicating flawless classification. Despite larger training datasets, the classifier's performance on the original TCGA+DKUH datasets did not significantly increase. The classifier's poor generalizability to outside datasets was significantly improved upon when it was retrained using a combination of tissue imaging data from TCGA, DKUH, and SSMH [2].

The capacity of deep learning, a subtype of machine learning, to educate computer systems to learn by illustration is highlighted, along with its applications in fields including object recognition, disease detection, and driverless cars. Deep learning's numerous uses are described, demonstrating its broad influence on a variety of industries. These include automated driving, aerospace and defense, medical research, industrial automation, and electronics. The document also includes references to a number of studies and research publications on the analysis of gene expression data, the categorization of cancer, and the application of gene expression data to the identification of cancer cells and the prognosis of cancer patients. It highlights how important deep learning and gene expression are to raising the bar for technology and producing hitherto unachievable outcomes [3].

The discovery of strongly associated gene modules in TCGA-LIHC—the turquoise, brown, and blue modules in particular—and their relationship to the emergence of tumors and the progression of liver hepatocellular carcinoma are covered in this paper. It draws attention to the function of genes in the turquoise module, which are abundant in KEGG pathways linked to the onset and spread of liver cancer as well as GO–Biological Process keywords pertaining to the apoptotic process and positive control of anoikis. To investigate the highly associated genes and their functional importance, the study also used WGCNA, GO, and KEGG pathway enrichment analysis. The report includes findings from investigations of the KEGG pathway, Reactome, and immune cell infiltration levels [4].

The article covers several feature selection techniques as well as next-generation sequencing technologies for gene expression data analysis and cancer classification. It discusses how to use machine learning algorithms, Bayesian techniques, and deep learning to find biomarkers and categorize various cancer kinds using RNA-Seq data. The significance of molecular cancer classifications such as those for clear cell sarcoma and triple-negative breast cancer for enhancing patient selection for treatment and early cancer detection is emphasized in the document. It emphasizes the application of machine learning algorithms and numerous feature selection techniques for reliable biomarker identification in lung cancer and hepatocellular carcinoma subtypes. The use of class specific feature selection for cancer detection and the analysis of gene expression profiles of lung cancer subtypes are also covered in the document [5].

Based on gene expression data, several machine learning and deep learning methods have been created for the identification and classification of cancer. Several methods have been proposed for accurate cancer categorization, including ensemble models, deep neural networks, boosting cascade deep forest models, and genetic algorithms. To improve the performance of the models, methods such as dimensionality reduction, feature selection, and transfer learning have been applied. The models are designed to display molecular clusters, forecast cancer survival using structured input data, and discover genetic flaws causing cancer growth. With a focus on their potential for precise classification and diagnosis, the paper offers a thorough assessment and comparative analysis of the machine learning and deep learning models utilized for cancer detection [6].

The research demonstrates improved cancer prognosis accuracy compared to conventional approaches by ensembling data from multiple classifiers and using deep learning-based methods. The study focuses on lung, stomach, and breast adenocarcinomas and explores the application of deep learning techniques in enhancing the accuracy of cancer prediction models. The study also examines the role of statistical methods in selecting informative genes and analyzes gene expression data to provide more accurate and effective cancer diagnosis. Finally, the research highlights the potential of deep learning in improving cancer detection and treatment decision-making [7].

The importance of deep learning techniques for handling large data and obtaining high prediction accuracy is covered in the paper. It draws attention to the shortcomings of conventional machine learning techniques in managing unprocessed genomic data and underscores the possibilities of deep learning in identifying trends and advancing the field of genomics. It highlights how deep learning techniques have the power to transform artificial intelligence and advance genomics research, which will help industries like pharmacy, agriculture, and medicine. The paper is divided into sections that include the use of deep learning architectures in genomic research, developments in deep learning architecture, the constraints of deep learning models in genomic research, and a general overview of deep learning models [8].

The Cancer LivER architecture is thoroughly explained in the article, along with the several packages that are utilized for data processing and visualization. The paper highlights the extraction of gene-related data from Enrichr and other databases, the integration of analysis modules, and dataset preprocessing. CancerLivER aims to assist researchers in selecting genes for use as disease biomarkers and treatment targets. It also emphasizes how important it is to regularly add new genomic features, along with RNA expression data, to the database. The database contains information on biomarkers from multiple sources as well as clinical research related to liver cancer. CancerLivER offers solutions for data retrieval and analysis to facilitate researchers' efficient access to and use of the database [9].

The proposed CNN architecture and its efficiency in processing high-dimensional data are the main topics of the document, which addresses the application of a convolutional neural network (CNN) for cancer classification. The proposed LS-CNN model is found to function optimally, as evidenced by its high classification accuracies (ranging from 0.90 to 1) across multiple datasets. Showcasing the precision and effectiveness of the model is a thorough examination of the best outcomes for every dataset, complete

with confusion matrix, elapsed time, and number of iterations. This research highlights the competitive performance of the LS-CNN model based on classification accuracy values by comparing it with other methods currently in use. The accuracy graphs for the several datasets are displayed, showcasing the consistency of high accuracy values. The accuracy of 0.938 is slightly lower in one dataset, though it is still present [10].

Numerous research projects are devoted to the diagnosis, treatment, and early detection of cancer, which is a major global health concern. With an emphasis on utilizing big data, such as transcriptome profiling from The Cancer Genome Atlas (TCGA), deep learning (DL) has become a potent tool for cancer prediction and detection. Many deep learning models have been created and used to accurately categorize and predict various cancer kinds. These models include convolutional neural networks (CNN), k-nearest neighbors (KNN) algorithm, and fully connected deep neural networks (DNN). The report emphasizes the significance of accurately classifying the tissue of origin within each form of cancer and the necessity of systematically assessing various CNN model architectures [11].

Soft tissue sarcoma subtypes were subjected to differential gene expression analysis, which produced unique molecular profiles and putative diagnostic indicators. Every subtype has GO keywords linked to it via enrichment analysis, which highlights variations in biological processes including cell cycle regulation and muscle differentiation. Three separate groups were formed from the subtypes by t-SNE analysis, which was based on shared morphological traits and molecular profile similarities. With the exception of differentiating between MFS and UPS, a random forest machine learning technique produced subtype predictions with above 0.95 accuracy. Significant subtype-defining genes were found, including MDM2 and CDK4 for DDLPS, and HOXA11 and its antisense RNA for STLMS and ULMS [12].

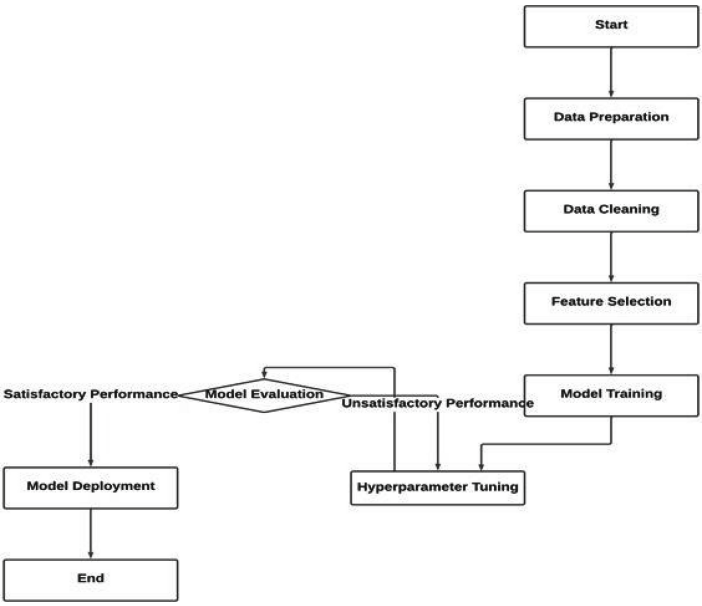
Large-scale phase III trials have explored a number of targeted therapies for hepatocellular carcinoma (HCC), but none have shown to be more effective than sorafenib. A strong three subclass structure was found in the molecular classification of human HCC based on gene expression profiles from diagnostic biopsy specimens, which was linked with both clinical characteristics and overall survival. Previous HCC molecular classification techniques were only relevant for individuals with advanced HCCs since they depended on HCCs that had been surgically removed. Using gene expression profiles from diagnostic biopsy specimens, the study attempted to create a molecular categorization of human HCC. It found that majority of the" signature" genes that define subclasses were differently regulated in the same direction in all cancers [13].

The application of Support Vector Machines (SVM) to cancer classification and the difficulties of gene array-based cancer classification are covered in the research. Many methods of gene selection and ranking are discussed, such as the application of SVM and the Wilcoxon rank sum test for gene selection in cancer classification. The research emphasizes how crucial it is to find marker genes for cancer detection and how to choose gene subsets using tabu search (TS). AML and ALL are two examples of identical tumors that can be distinguished from one another using different methods, with an emphasis on the function that neural networks play in classification. An analysis is conducted on the effectiveness of the NPPC ensemble and its combination with SVM for cancer

diagnosis; the results indicate testing accuracy comparable to that of the SVM ensemble with shorter training times [14].

3 Proposed Model

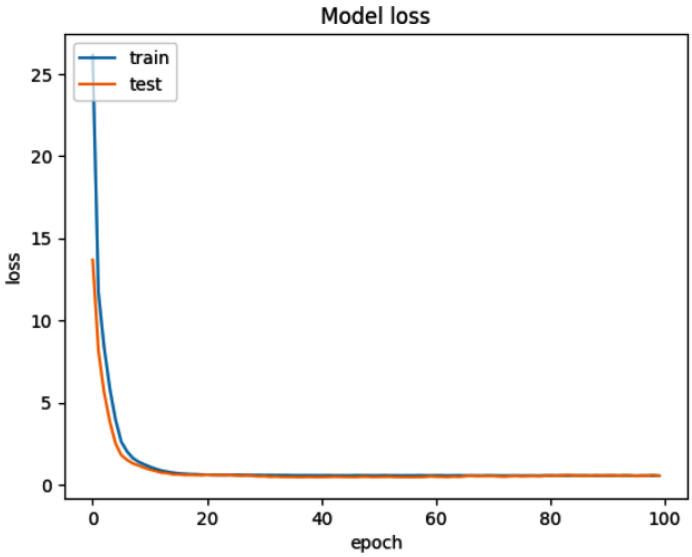
1. Data Collection: Specific libraries must be imported for plotting, data processing, and machine learning. A Pandas Data Frame is created by loading a CSV file containing the dataset.
2. Data Preprocessing: All missing values are filled with 0. The one-step technique converts the'Gender' column to numerical values, resulting in the'Gender-Female' and'Gender-Male' columns. Renaming the'Dataset' column leads to the following modifications. The patient with liver becomes 0, whereas the one without liver becomes one.
3. Feature Selection and Extraction: The only columns used as input characteristics are 'Dataset' and 'Gender'. The 'Dataset' column is transformed into category labels using one-hot encoding, resulting in two classes: liver patient (0) and non-liver patient (1).
4. Neural Network: The model comprises the following layers: The input layer consists of ten neurons in a thick layer activated by ReLU. The hidden layer consists of eight neurons and uses ReLU activation. The output layer includes sigmoid activity and two neurons in a thick layer.
5. Testing and Prediction: The model is built using diminished. The function is named binary cross-entropy. Adam is the optimizer. Metrics: Precision. Training the model requires 100 epochs, with ten in each batch. Validation takes up 20% of the data.
6. Model Evaluation: Using the training data, the model's accuracy and loss are analyzed. To understand the parameters and structure, the model architecture is summarized.

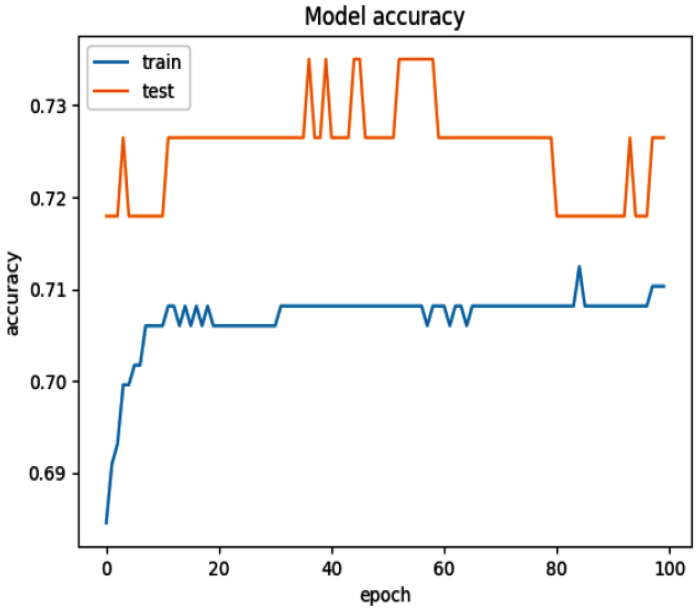


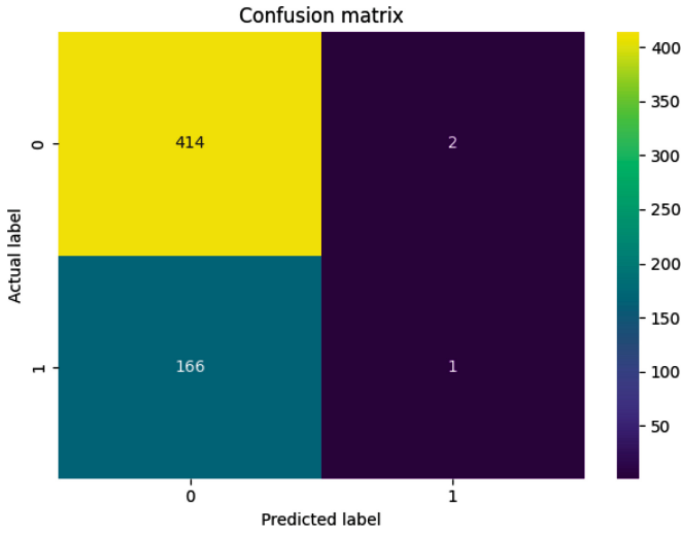
The proposed architecture is depicted in the flowchart. To make sure the dataset is reliable and consistent, it starts with the data preparation stage and continues with data cleaning. The following stage is feature selection, in which pertinent features are picked so that the model can be trained. The model's performance is assessed after training. The model is implemented if the performance is deemed acceptable. If not, the model's performance is enhanced through hyperparameter adjustment. This cycle of training, assessment, and fine-tuning doesn't stop until the model performs well enough to be deployed, marking the end of the process.

4 Results and Discussion

The proposed model shows how accurate a neural network model is over training epochs. Using a patient data dataset, the suggested method constructs and assesses a neural network model for liver disease classification. Following the conversion of categorical variables and preprocessing of the data, the model is trained with many layers and its performance evaluated. The model's performance is assessed using a variety of measures, including sensitivity, specificity, precision, and the confusion matrix. The training method yields a reasonable standard of accuracy. The learning process over epochs is depicted by the accuracy and loss plots, while the confusion matrix demonstrates the model's capacity to discriminate between examples that are ill and those that are not.







In Summary, the model's accuracy is roughly 0.672, precision is 0.452, recall is 0.671, and its F1 score is 0.539. These metrics indicate that the model is somewhat effective, with recall leading precision.

5 Conclusion

Most cases of liver cancer are curable. 16 months is the median survival time for liver cancer. With transarterial chemoembolization, it could extend beyond 40 months (TACE). With treatment, the median survival is 11–13 months; in the absence of treatment, it is 6–8 months. When they do manifest, symptoms and indicators could include inadvertent weight loss, appetite loss, upper abdomen pain, nausea, vomiting, overall weakness, and exhaustion. This project shows how to use patient data to construct and assess a neural network for the purpose of classifying liver disease. The neural network is trained to differentiate between individuals with and without liver disease by pre-processing the data, generating features, and assigning labels to the data. The model is assembled and trained over a hundred epochs and consists of layers with activation functions. Metrics like loss, accuracy, sensitivity, specificity, and precision demonstrate its correctness to some extent. The model's performance is graphically represented by the confusion matrix, which shows true positives, true negatives, false positives, and false negatives. All things considered, the neural network exhibits promise in the classification of liver illness; accuracy and other metrics offer a thorough assessment of its effectiveness.

6 Future Scope

In the future we would explore the potential of combining neural networks (NNs) with other machine learning techniques for improved performance. Techniques like dropout, regularization, or more complex architectures (like CNNs or RNNs) may be required in order to enhance the model. Accuracy in training and validation may also be stabilized and improved with additional data or data augmentation. While we showed that neural networks can be used to classify liver illness, further development can increase the system's accuracy and performance. It is possible to apply explainable AI techniques to comprehend how neural networks (NNs) arrive at their classifications. Eventually consider the potential for integrating the neural networks (NNs) model into clinical practice for earlier and more accurate liver cancer diagnosis.

For comparison, advanced approaches like support vector machines (SVMs) and other machine learning algorithms like ensemble learning (e.g., Random Forests, Gradient Boosting) could be investigated. The robustness and generalizability of the model might be further confirmed by applying cross- validation techniques and growing the dataset with more balanced and diverse samples, which would ultimately result in improved diagnostic instruments for the classification of liver disease. The proposed methodologies should be included to the model, which produces a precise medication, to improve system efficiency. Overall, even though the model's accuracy is respectable, more fine-tuning and testing are necessary to guarantee reliable and broadly applicable performance.

References

1. Zhao, Z., et al.: A machine learning method for improving liver cancer staging. J. Biomed. Inform. **137**, 104266 (2023)
2. Jang, H.-J., et al.: Deep learning for the pathologic diagnosis of hepatocellular carcinoma, cholangiocarcinoma, and metastatic colorectal cancer. Cancers **15.22**, 5389 (2023)
3. Retracted: gene expression-assisted cancer prediction techniques. J. Healthcare Eng. 1 (2023). https://doi.org/10.1155/2023/9858247
4. Chen, Y., Huang, W., Ouyang, J., Wang, J., Xie, Z.: Identification of anoikis-related subgroups and prognosis model in liver hepatocellular carcinoma. IJMS **24**(3), 2862 (2023). https://doi.org/10.3390/ijms24032862
5. Alharbi, F., Vakanski, A.: Machine learning methods for cancer classification using gene expression data: a review. Bioengineering **10**(2), 173 (2023)
6. Sumalatha, G., Muthumani, D.N.: Gene expression analysis for cancer prediction: a review of machine learning and deep learning techniques no. 7 (2023)
7. Himavarnika, A., Prasanthi, P.: A statistical modelling to detect carcinoma cancer in its incipient stages in healthcare. J. Coastal Life Med. **11**, 468–481 (2023)
8. Abass, Y.A., Adeshina, S.A.: Deep learning methodologies for genomic data prediction: review. JoAIMS **2**(1–2), 1–11 (2021). https://doi.org/10.2991/jaims.d.210512.001
9. Kaur, H., et al.: CancerLivER: a database of liver cancer gene expression resources and biomarkers. Database 2020 (2020): baaa012
10. Shah, S.H., et al.: Optimized gene selection and classification of cancer from microarray gene expression data using deep learning. Neural Comput. Appl. 1–12 (2020)
11. Mostavi, M., et al. "Convolutional neural network models for cancer type prediction based on gene expression. BMC Med. Genomics **13**(2020), 1–13 (2020)
12. van IJzendoorn, D.G.P., et al.: Machine learning analysis of gene expression data reveals novel diagnostic and prognostic biomarkers and identifies therapeutic targets for soft tissue sarcomas. PLoS Comput. Biol. **15.2**, e1006826 (2019)
13. Makowska, Z., et al.: Gene expression analysis of biopsy samples reveals critical limitations of transcriptome-based molecular classifications of hepatocellular carcinoma. J. Pathol. Clin. Res. **2.2**, 80–92 (2016)
14. Rajeswari, P., Reena, G.S.: Human liver cancer classification using microarray gene expression data. Int. J. Comput. Appl. **34.6**, 25–37 (2011)

Optimizing Tomato Leaf Disease Identification Using a Hybrid Spatial-Temporal Model and Attention Mechanism

Sajeev Ram Arumugam[1(✉)], P. Sheela Gowr[2], E. Anna Devi[3], J. Elavarasi[4], and Sankar Ganesh Karuppasamy[5]

[1] Department of AI&DS, Sri Krishna College of Engineering and Technology, Coimbatore, India
imsajeev@gmail.com
[2] Department of CSE, Vels Institute of Science Technology and Advanced Studies, Chennai, India
[3] Department of ECE, Sathyabama Institute of Science and Technology, Chennai, India
[4] Department of AI&DS, Vel Tech High Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Chennai, India
[5] Department of CSE, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, India

Abstract. Ensuring crop health and yield requires accurately identifying diseases in tomato leaves. Existing deep learning models may not adequately capture the spatiotemporal patterns of disease progression, and traditional methods can be laborious and subjective. The hybrid spatial-temporal model enhanced by the attention processes used in this study presents an optimal method for diagnosing tomato leaf disease. The proposed approach uses DenseNet to extract complex spatial characteristics from high-resolution leaf images and recurrent neural networks (RNNs) to predict the temporal course of illness symptoms. The integration of temporal and spatial data provides a solid framework for classifying diseases. The model may also selectively focus on important areas of the images by using attention mechanisms, which improves classification accuracy. The objective is to develop a robust and efficient system for early disease detection in tomato crops. A key challenge lies in ensuring the generalizability to complex real-world scenarios with diverse disease presentations. Comprehensive evaluations of a wide range of datasets show that our hybrid model far outperforms traditional techniques in identification performance, underscoring its potential for real-world use in precision agriculture.

Keywords: Tomato Leaf Disease · Classification · Spatial-Temporal Fusion · DenseNet · Attention Mechanism · Deep Learning · Precision Agriculture

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 99–110, 2025.
https://doi.org/10.1007/978-3-031-86296-0_8

1 Introduction

A major agricultural risk, tomato leaf diseases specifically impact tomato production, a product vital for personal gardening and marketable farming. The diseases can considerably lower tomato crop yield and quality since various pathogens, such as bacteria, viruses, and fungus, bring them on [1]. Tomatoes are one of the most commonly grown vegetables in the world, and the impact of these diseases is important. Accurate disease classification allows farmers to handle diseases early, evading their spread and decreasing crop damage. Rapid disease propagation resulted in a path of damaged leaves and, eventually, massive crop losses. It results in a significant financial setback for growers and may cause problems for the world's tomato supply chain. Fighting back and preserving priceless tomato crops depends on the early and precise detection of these diseases. It allows for early diagnosis and execution [2]. This preventive strategy promises higher yields and healthier crops. In addition to decreasing input costs, this reduces environmental impact by avoiding chemical exploitation.

The most common tomato leaf diseases are Brown dots that appear on leaves, stems, and fruit due to early blight, a fungal disease. The patches may eventually cause the leaves to die from growing together. Big, water-soaked sores on leaves, stems, and fruit result from the fungus known as late blight. The fruit may decay, and the blemishes can soon turn dark and fuzzy. Small, circular leaf patches are caused by the fungus known as septoria leaf spot. The patches may eventually cause the leaves to die from growing together [3]. Fruit and leaves may get tiny, wet patches on them due to an infection caused by bacteria called "bacterial spot." The patches may eventually cause the leaves to die from growing together. Inedible rotting fruit is another possibility. Tomato Spotted Wilt virus causes reduced leaf development, yellowing, and bronzing and is spread by thrips. The fruit may also be deformed and discolored. Figure 1 shows the examples of tomato leaf diseases.

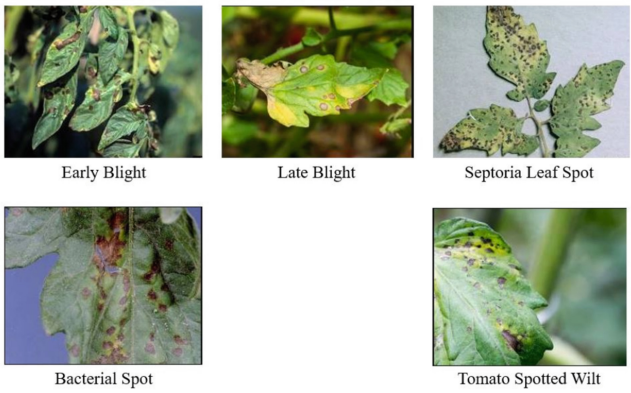


Fig. 1. Tomato Leaf Diseases

The challenges in tomato leaf disease classification are:

- Visual Similarity: Differentiating between several tomato leaf diseases can be challenging due to their similar visual symptoms.
- Environmental Factors: Different growth steps, weather patterns, and lighting conditions can all impact how diseases manifest.
- Data Variability: Diverse datasets resolutions, annotations, and quality may vary.

This research aims to develop a method for identifying and classifying diseases of tomato leaves using a hybrid spatial-temporal model with attention mechanisms. The proposed model aims to attain visual patterns by performing spatial and temporal analysis on individual leaf images and sequences of images. Furthermore, attention mechanisms will emphasise the model's attention on significant leaf regions, resulting in a more accurate and reliable disease detection system [4]. This work is significant because it offers accurate disease classification, which helps with precision farming. Specific approaches are made possible by early and accurate detection of tomato leaf diseases, reducing the need for wide-ranging pesticides and allowing more environmentally friendly solutions. Furthermore, by classifying the exact disease, farmers may increase output potential and enhance resource allocation by concentrating on the affected areas. The vital goal of the research is to develop precision agriculture, which will reinforce the food supply chain and increase tomato production in the prospect.

2 Literature Review

Every year, tomato leaf diseases lead to large crop losses and loom the world's food security. The employment of control measures and the depreciation of financial significance rely upon the early identification of these diseases. Traditional methods rely on the subjective, time-consuming, and arduous visual evaluation of specialists. The purpose of this review of the literature is to examine existing developments in deep learning-based automated tomato leaf disease identification. Researchers considered the perspective of deep learning to detect tomato leaf diseases. They assessed CNN models that were previously trained, such as ResNet and MobileNet, and were able to categorize diseases with high accuracy, up to 99.81%. Optimization strategies for smaller models appropriate for hardware with low resources were observed in the study. A Raspberry Pi 4's successful employment shows probable for agricultural applications using Internet of Things gadgets. The study does not, however, define the diseases the models can classify, and more research is crucial to see whether the models can manage a variety of disease presentations in real-world situations [5].

This study examined how tomato leaf diseases could be categorized using a deep-learning architecture called AlexNet. They were highly accurate when detecting different tomato leaf diseases using images (around 98%). It demonstrates AlexNet's ability to understand visual patterns linked to various diseases efficiently. The number of disease classifications the model could distinguish between is not stated in the paper. Although the model's accuracy is outstanding, it's uncertain if it can handle the wide range of tomato leaf diseases that arise in farming environments [6]. The study proposes FC-SNDPN, a new technique for identifying tomato leaf disease. This model efficiently

distinguishes between healthy and diseased leaves with a high accuracy rate of 97.59% [7]. However, the absence of specificity regarding the diseases the model may identify is a major flaw.

This study presents a deep learning method for accurate tomato leaf disease identification in images. PLPNet has a Proximity Feature Aggregation Network to handle diseases with similar appearances, a Location Reinforcement Attention Mechanism to emphasise prospective disease areas, and a Perceptual Adaptive Convolution module to excerpt disease-specific details. PLPNet obtained 94.5% mean average precision with 50% thresholds (mAP50), 54.4% average recall (AR), and 25.45 frames per second (FPS) [8]. However, concerns have been raised regarding the model's capacity to handle numerous real-world measures. Furthermore, architecture could need a lot of processing power, which might stop it from being used in agriculture. They investigate the DenseNet121, ResNet50V2, and ViT models on test data, attaining great accuracy of up to 99.88% [9]. It validates how well these deep-learning models classify diseases. The research does not mention integrating the transform methods into the complete model. The results of existing research on deep learning for tomato leaf disease detection have been encouraging; this work aims to address some important areas that still need development:

- Attention Mechanisms: An increasing number of models are adopting attention mechanisms to improve performance by concentrating on the most relevant portions of the input images [10].
- Combination of Spatial Data: Efforts have been made to include spatial data into disease identification models, such as leaf position and disease propagation patterns [11].
- Combination of Temporal Data: Individual leaf images are the subject of most research. Through the analysis of image sequences, this work will examine the combination of temporal information to track the growth and maybe identify them at an earlier stage [12].

The proposed work presents a novel way to detect tomato leaf disease. A hybrid spatial-temporal model is presented in contrast to existing works. This model will inspect a time series of images to screen the disease's growth and discrete leaf images to identify disease trends. Furthermore, sophisticated attention mechanisms are used to screen the change of disease patterns through image sequences and focus on important areas within leaves. Additionally, we prioritise generalizability by using methods to handle complex backgrounds and diverse lighting conditions and by training the model on a wider spectrum of diseases. The goal of this proposed approach is to develop a disease detection system that is more consistent and precise, and that can be used in agricultural environments.

3 Methodology

Ensuring crop health and production necessitates diagnosing diseases affecting tomato leaves quickly and precisely. Although deep learning models are a good substitute, they won't capture the randomness of disease development over time. This study proposes

a unique tomato leaf disease identification method that uses a hybrid spatial-temporal model with attention mechanisms. This section explores the methodology used to create and assess this new model.

3.1 Dataset Description

Researchers working on machine learning-based automated plant disease identification find the PlantVillage dataset revolutionary. This open-source database provides an enormous number of images, usually—that show healthy and sick leaves from various plant species [13]. Training strong models to address a broad disease spectrum requires this diversity. The PlantVillage dataset is a key resource for research on automated plant disease identification, despite the possibility that certain versions do not have comprehensive annotations of particular disease types on the leaves. This is because of the dataset's large size and general concentration on leaves. In the proposed work, 18279 images are used for training and 4568 images for testing.

Data augmentation can greatly improve the quality and diversity of tomato leaf images when identifying tomato leaf diseases. The color space Augmentation technique [14] regulates brightness and contrast. To produce miscellaneous lighting conditions, modify the images' brightness and contrast. It aids the model in doing well in various lighting conditions. Data augmentation retains the model from overfitting to the training set by generating a more varied dataset. As a result, the model works better with unknown data. Color space segmentation converts the image into a color space that more effectively distinguishes affected regions, assisting in identifying diseased areas.

Convert the image I to a chosen color space using Eq. (1).

$$I_{HSV} = RGBtoHSV(I) \tag{1}$$

Segmenting the image using thresholds in the HSV color space using Eq. (2):

$$M_{HSV}(x, y) = \begin{cases} 1 \ if \ H_{min} \leq H(x, y) \leq H_{max} \ and \ S_{min} \leq S(x, y) \leq S_{max} \\ 0 \ otherwise \end{cases} \tag{2}$$

Where H(x,y) and S(x,y) are the hue and saturation values. M_HSV is the binary mask highlighting potentially diseased areas. This mask can recognise regions of interest (ROI) in the image.

3.2 Spatial-Temporal Model

The proposed approach for identifying tomato leaf diseases uses a hybrid architecture combining an attention mechanism with spatial and temporal analysis benefits. Convolutional neural networks (CNNs) are used in the spatial model to capture the spatial characteristics of tomato leaf disease in a single image. In the proposed approach, CNN architecture DenseNet is used for image classification. DenseNet (Densely Connected Convolutional Networks) reduces the parameters and provides maximum feature reuse by creating connections between each layer. DenseNet efficiently extracts complex spatial characteristics from images to diagnose tomato leaf disease. In DenseNet, each layer

takes input from all preceding layers. If there are L layers, the output layer Xl I defined as in Eq. (3)

$$X_l = H_l\left(\left[X_0, X_1, \ldots\ldots..X_{l-1}\right]\right) \tag{3}$$

Where [X_0,X_1,……..X_(l-1)] denotes the concatenation of feature maps from all previous layers, H_l is a composite function of operations. A dense block comprises several levels, each of which has as its input the concatenation of the outputs from all its preceding layers as in Eq. (4)

$$X_l = concat\left(\left[X_0, X_1, \ldots\ldots..X_{l-1}\right]\right) \tag{4}$$

Transition layers decrease the number of channels and the size of the feature maps between dense blocks as in Eq. (5)

$$X_{trans} = BN\left(RELU\left(Conv\left(X_{dense}\right)\right)\right) \tag{5}$$

The final dense block output, Fspatial, represents rich spatial features. Features from the image, such as textures, shapes, and color patterns, are extracted through convolutional layers. The feature maps are down sampled using pooling layers, which depresses the dimensionality while keeping vital information. Recurrent neural networks (RNNs) are used in the temporal model to study the disease's temporal features. This part is useful if the dataset contains several images of the same leaf taken at different times. It helps the model to identify developments in the disease. When processing sequential data, RNNs surpass at capturing how disease features change throughout the image series. Given a sequence of spatial features Fspatial, the temporal model works as shown in Eqs. (6) to (11).

$$Input\ GateI_t = \sigma\left(W_i.\left[h_{t-1}, F_{spatial}\right] + b_i\right) \tag{6}$$

$$Forget\ Gatef_t = \sigma\left(W_f.\left[h_{t-1}, F_{spatial}\right] + b_f\right) \tag{7}$$

$$Cell\ State\ \tilde{C}_t = tanh\left(W_c.\left[\left[h_{t-1}, F_{spatial}\right] + b_c\right]\right) \tag{8}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{9}$$

$$Output\ GateO_t = \sigma\left(W_o.\left[h_{t-1}, F_{spatial}\right] + b_o\right) \tag{10}$$

$$h_t = O_t \odot tanh(C_t) \tag{11}$$

Here σ denotes the sigmoid function, Θ represents element-wise multiplication, and h_t is the hidden state at time t. The final output, Ftemporal, is obtained from the sequence of hidden states. Figure 2 shows the working of the temporal model.

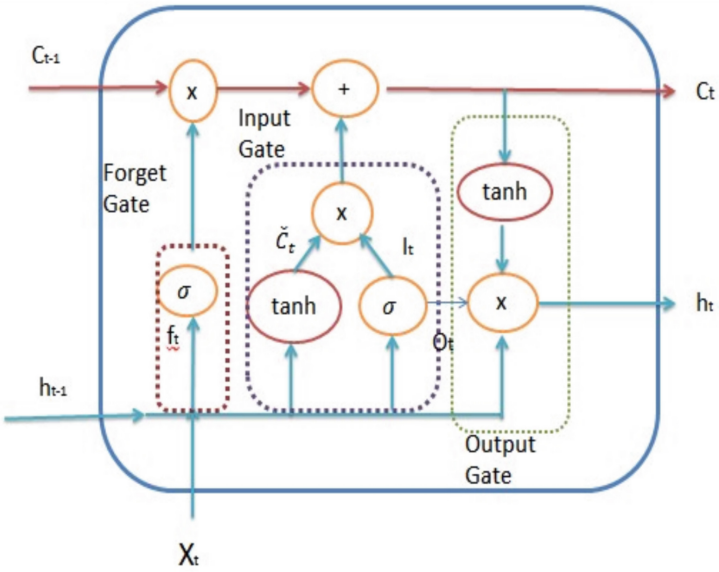


Fig. 2. Temporal Model

3.3 Attention Mechanism

An effective combination of the features collected from the temporal model (RNN) and the spatial model (CNN) is attained. This fusion allows the model to gain temporal knowledge about the disease's progression and spatial information about the current state of the disease. The architecture of the model integrates an attention mechanism. This method guides the model's attention to the most informative sections of the image sequence, which are acute for identifying diseases. It is useful for early-stage diseases. Finally, the disease class for the tomato leaf is predicted using a classification layer with softmax activation. Because of the benefits of merging temporal and spatial data, the model may outperform models that only use spatial variables and develop a deeper understanding of disease features. Figure 3 shows the workflow of the proposed architecture.

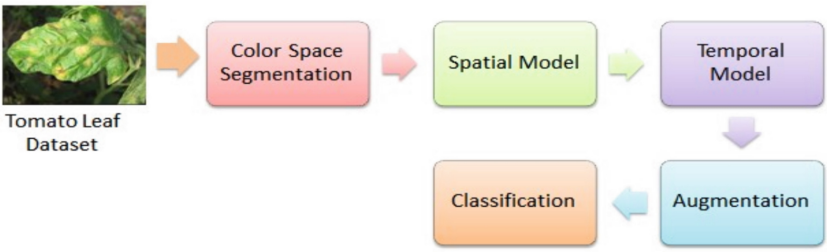


Fig. 3. Workflow of the proposed architecture

The most significant regions in an image are the centre of spatial attention. The attention mechanism computes an attention map A for an input feature map Fspatial. Attention Scores compute scores for each spatial location (i,j) as in Eq. (12).

$$\propto ij = \frac{exp(eij)}{\sum_{k,l} exp(ekl)} \tag{12}$$

Where (*eij*) represents the score Fspatial for location (i,j). The attended feature map can be obtained by multiplying the attention scores by the original feature map as in Eq. (13).

$$F_{att} = \sum_{i,j} \propto ijF_{spatial_{i,j}} \tag{13}$$

Temporal attention focuses on important time steps within a sequenceas in Eq. (14).

$$\beta_t = \frac{exp(u_t)}{\sum_{k=1}^{T} exp(u_k)} \tag{14}$$

Where u_t= score(ht) is the relevance score for time step t, typically computed as a function of the hidden state at that time step. The context vector of the attended sequence can be obtained by calculating the weighted sum of the hidden states as in Eq. (15).

$$c = \sum_{t=1}^{T} \beta_t h_t \tag{15}$$

One possibility is to use a hybrid mechanism to integrate temporal and spatial attention. Each frame's feature map should receive separate spatial attention as in Eq. (15).

$$F_{t,att} = SpatialAttention(F_t) \tag{16}$$

Apply temporal attention to the series of spatially attended features, taking into account framestemporal attention to the series of spatially attended features, taking into account frames as in Eq. (16):

$$C_{final} = \sum_{t=1}^{T} \beta_{t,att} \tag{17}$$

Important areas within each frame and significant frames within the sequence are the focus of this integrated approach.

4 Performance Evaluation

Performance evaluation in tomato leaf disease identification involves several rigorous steps to correctly endorse the model's capacity to identify various disease stages from leaf image sequences. This process often begins with the dataset split into training and test sets to aid objective evaluation. Table 1. Shows the number of training and testing samples for each type of disease.

Table 1. Training and Testing samples for every class

Class	Number of training samples	Number of testing samples
Early Blight	2723	680
Late Blight	3821	955
Septoria Leaf Spot	4729	1182
Bacterial Spot	1384	346
Tomato Spotted Wilt	3828	957
Healthy Leaf plant	1794	448
Total	**18,279**	**4568**

Developing and implementing machine learning models requires careful consideration of performance evaluation, particularly for applications like tomato leaf disease detection, where accurate and reliable predictions can greatly impact agricultural results. Estimating the model's ability to simplify to new, unknown data and confirming that it satisfies the necessary accuracy and robustness requirements for practical deployment are the main objectives of performance evaluation. Developing a productive model for tomato leaf disease detection requires careful consideration of hyperparameter tuning and optimization. The objective is to identify the ideal hyperparameter combination that improves the model's performance on simulated data. The learning rate is 10-6, which determines the number of samples processed before the model's parameters are updated; the batch size is 32; and the number of layers and units in the DenseNet and RNN architectures is 201, which affects the model's ability to learn spatial and temporal features.

The percentage of accurately categorized instances (healthy and diseased) relative to the total number of instances is known as accuracy. It is an overall measure of how frequently the model predicts its outcomes accurately.

$$Accuracy = \frac{Number\ of\ Accurate\ Findings}{Total\ Number\ of\ Findings} \tag{18}$$

Sensitivity is the percentage of actual diseased leaves that the model accurately detects. It is essential to ensure the model captures most sick incidents.

$$Sensitivity = \frac{TP}{TP + FN} \tag{19}$$

Where TP is the number of diseased leaves correctly identified by the model, FN is the number of diseased leaves incorrectly identified as healthy.

A model's specificity indicates its accuracy in detecting negative situations. Specificity in tomato leaf disease detection refers to how well the model distinguishes healthy leaves so they are not mistakenly labeled as diseased.

$$Specificity = \frac{TN}{TN + FP} \tag{20}$$

Where TN is the number of healthy leaves correctly identified as healthy by the model, FP is the number of healthy leaves incorrectly identified as diseased. The system achieves an accuracy of 0.9863, a sensitivity of 0.9737 and a specificity of 0.9715. The number of epochs is set to 75. The Loss vs. Epochs graph is an effective tool for analyzing and showing how a machine-learning model is being trained. In particular, in applications like tomato leaf disease detection, where accurate and robust models are critical, one can make accurate decisions about model changes and optimizations to obtain greater performance and generalization by regularly monitoring this graph. Figure 4 shows the expected loss value of the training and test datasets in the learning process.

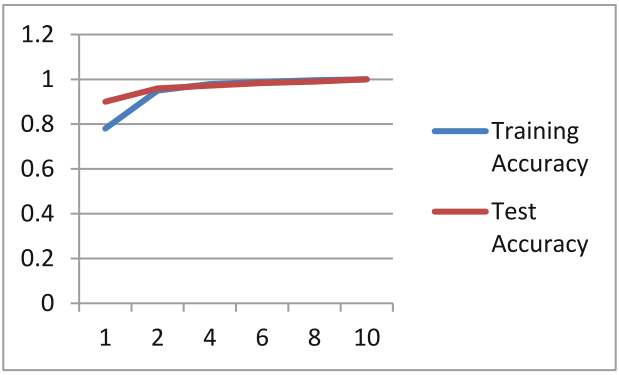


Fig. 4. Loss Vs Epochs

An essential tool for evaluating the model's performance over time is the Accuracy vs. Epochs graph. This graph shows how the model's capacity to identify healthy and diseased leaves changes during training by plotting the accuracy on the y-axis against the number of epochs on the x-axis. A situation known as overfitting is one in which the model exhibits good training accuracy on observed data. It is typically graphed and studied in the training process, and the value is frequently related to the complete model accuracy, as shown in Fig. 5.

The proposed tomato leaf disease identification method is compared with the existing methods and shown in Table 2.

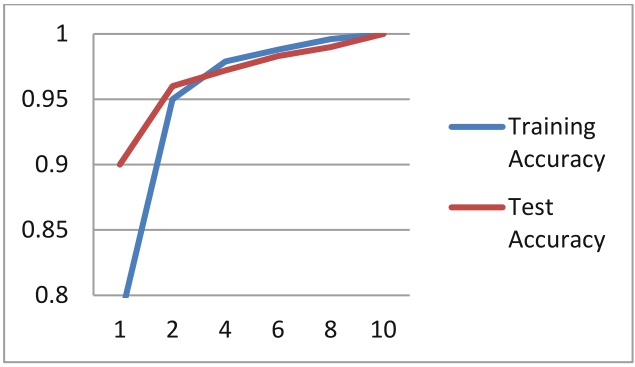


Fig. 5. Accuracy Vs Epochs

Table 2. Comparative Analysis

Author	Techniques	Accuracy	Sensitivity	Specificity
Anandhakrishnan [15]	DCNN	98.4%		
Hsing-Chung Chen [6]	AlexNet CNN	98%	98%	96%
Xibei Huang [7]	FC-SNDPN	97.59%		
Alaa Saeed [16]	Transfer Learning and CNN	99.22%		
Proposed Work	Spatial-temporal and attention mechanism	98.63%	97.37%	97.15%

5 Conclusion

In this research, we proposed a hybrid model that combines attention mechanisms, recurrent neural networks (RNN), and DenseNet. It offers a novel method for capturing temporal and spatial patterns in images of tomato leaves. The proposed approach for identifying tomato leaf disease performs remarkably well, with an overall accuracy of 98.63%. The model is highly trustworthy for real-world agricultural applications since it accurately detects damaged and healthy leaves. Furthermore, the model's 97.37% sensitivity demonstrates its excellent ability to identify infected leaves, essential for rapid action and stopping the spread of diseases. It reduces the possibility of undiagnosed infections by ensuring that almost all disease cases are correctly identified. Likewise, the model attains a 97.15% specificity, demonstrating its ability to adapt in accurately categorizing healthy leaves and preventing false positives. Preventing healthy plants from receiving needless treatments is crucial for preserving the effectiveness of disease management. The model is a useful tool for improving crop health and productivity; these measurements demonstrate its balanced and successful performance in identifying damaged leaves and verifying good ones.

References

1. Thangaraj, R., Anandamurugan, S., Pandiyan, P., Kaliappan, V.K.: Artificial intelligence in tomato leaf disease detection: a comprehensive review and discussion. J. Plant Dis. Prot. **129**(3), 469–488 (2022)
2. David, H.E., Ramalakshmi, K., Gunasekaran, H., Venkatesan, R.: Literature review of disease detection in tomato leaf using deep learning techniques. In: 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1, pp. 274–278 (2021). https://doi.org/10.1109/ICACCS51430.2021.9441714
3. Kibriya, H., Rafique, R., Ahmad, W., Adnan, S.M.: Tomato leaf disease detection using convolution neural network. In: 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST), pp. 346–351 (202)
4. Ram, S., Shylaja, C.S., Anandan, R.: Classification of air pollutants caused by e-waste and health risk evaluation. Ecol. Environ. Conserv. **26**(3), 1352–1361 (2020)
5. Tarek, H., Aly, H., Eisa, S., Abul-Soud, M.: Optimized deep learning algorithms for tomato leaf disease detection with hardware deployment. Electronics **11**(1), 140 (2022)
6. Chen, H.-C., et al.: AlexNet convolutional neural network for disease detection and classification of tomato leaf. Electronics **11**(6) (2022). https://doi.org/10.3390/electronics11060951
7. Huang, X., et al.: Tomato leaf disease detection system based on FC-SNDPN. Multimed. Tools Appl. **82**(2), 2121–2144 (2023)
8. Tang, Z., et al.: A precise image-based tomato leaf disease detection approach using PLPNet. Plant Phenomics **5**, 42 (2023)
9. Alzahrani, M.S., Alsaade, F.W.: Transform and deep learning algorithms for the early detection and recognition of tomato leaf disease. Agronomy **13**(5) (2023). https://doi.org/10.3390/agronomy13051184
10. Li, X., et al.: Deep learning attention mechanism in medical image analysis: basics and beyond. Int. J. Netw. Dyn. Intell. 93–116 (2023)
11. Marconato, L., et al.: SpatialData: an open and universal data framework for spatial omics. Nat. Methods 1–5 (2024)
12. Jin, G., et al.: Spatio-temporal graph neural networks for predictive learning in urban computing: a survey. IEEE Trans. Knowl. Data Eng. (2023)
13. Hughes, D.P., Salathe, M.: An open access repository of images on plant health to enable the development of mobile disease diagnostics, arXiv, 2015. http://arxiv.org/abs/1511.08060
14. Fadaeddini, A., Majidi, B., Souri, A., Eshghi, M.: Data augmentation using fast converging CIELAB-GAN for efficient deep learning dataset generation. Int. J. Comput. Sci. Eng. **26**(4), 459–469 (2023)
15. Anandhakrishnan, T., Jaisakthi, S.M.: Deep Convolutional Neural Networks for image based tomato leaf disease detection. Sustain. Chem. Pharm. **30**, 100793 (2022). https://doi.org/10.1016/j.scp.2022.100793
16. Saeed, A., Abdel-Aziz, A.A., Mossad, A., Abdelhamid, M.A., Alkhaled, A.Y., Mayhoub, M.: Smart detection of tomato leaf diseases using transfer learning-based convolutional neural networks. Agriculture **13**(1) (2023). https://doi.org/10.3390/agriculture13010139

Machine Learning Strategies for Proactive Malware Detection

Hardik Mahendrabhai Patel(✉) and Shyam Kalariya

Vidush Somany Institute of Technology and Research, Kadi, India
Profhmp9@gmail.com

Abstract. With the rapid evolution of network security threats, effective detection and analysis of malware has become important. This research focuses on new methods for malware detection and analysis using advanced machine learning algorithms. By integrating audited and unaudited programs, we aim to increase the accuracy and speed of malware detection. Our approach involves extracting and using different techniques from various malware samples; thus enabling the model to learn complex patterns and behaviors associated with malware. We investigate the effectiveness of different learning models in malware detection and distribution, including deep learning architectures, clustering and anomaly detection. Additionally, we examine the effectiveness of this model against evasion techniques used by sophisticated malware. Check performance in situations. This research helps develop more effective cybersecurity defenses and provides insights into the use of machine learning to combat malware threats.

Keywords: Machine Learning · Decision Tree · Cyber Security · Malware-detection · Thread

1 Introduction

In the digital age, the malware threat is evolving and poses a significant risk to individuals and organizations. Traditional malware detection methods often rely on signature matching and cannot keep up with the dynamic and evolving nature of modern malware. Therefore, there is always a need for more advanced detection tools that can predict and eliminate threats before they cause damage.

Machine learning holds promise for this challenge by using data-driven techniques to instantly detect and respond to crime. Unlike traditional methods, machine learning can analyze large amounts of data to identify subtle patterns and anomalies that may indicate the presence of malware. These techniques cover a wide range of methods, such as stealth detection, supervised learning, and deep learning, all of which contribute significantly to the malware detection task.

This paper takes an in-depth look at the application of machine learning in malware detection and examines how these techniques can be used to increase detection accuracy, reduce vulnerabilities, and adapt to emerging threats. By leveraging the power of machine learning, cybersecurity experts can develop stronger defenses that can identify and mitigate the impact of malware.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 111–119, 2025.
https://doi.org/10.1007/978-3-031-86296-0_9

In today's digital environment, the spread of malware attacks poses a threat to cybersecurity, affecting individuals, organizations, and government agencies. Traditional detection methods (only signature-based technology) often cannot keep up with the rapid evolution of malware. Therefore, more complex and adaptive detection mechanisms are urgently needed. Machine learning (ML) algorithms show promise against these challenges by using a data-driven approach to identify patterns and anomalies that indicate malicious behavior [4].

This work explores the development and use of machine learning algorithms to improve malware detection and analysis performance. We aim to increase the accuracy, speed and power of the malware detection process by performing supervised and unsupervised studies. Our approach involves removing useful values from malware models, making the model that describes complex behaviors and features specific to malware [2].

We evaluate the performance of various machine learning models, including deep learning models, hybrid algorithms, and error detection algorithms, in different scenarios. We also examined the effectiveness of this model against sophisticated evasion techniques used by advanced malware to ensure the reliability of existing crisis detection systems [3].

This study aims to demonstrate the potential of machine learning in improving malware detection capabilities through a comprehensive evaluation using real-world data. By analyzing the depth of various machine learning methods, we want to help make cybersecurity safer and ultimately improve the protection of assets and information [1].

2 Related Work

In recent years, significant advances have been made in malware detection and analysis using machine learning algorithms. Hou et al. (2021) proposed a new method called MalGraph, which uses graph-based deep learning for malware detection. Their research shows that graph neural networks (GNN) outperform traditional machine learning methods in detecting malware by capturing relationships in data. Salehi et al. (2022) focused on improving the resilience of malware detection models against attacks. Their research shows that by working on attack protection and differential networks (GANs), detection of defenses is improved and performance is guaranteed even with complex evasion techniques. Kumar et al. (2022) investigates the application of ensemble learning in malware classification. The results show that a combination such as random forest increases the accuracy and robustness of malware detection systems by combining the results of multiple distributions. Zeng et al. (2021) studied the integration of dynamic analysis method with machine learning to improve the zero-day malware detection rate. Their research shows that dynamic analysis combined with deep learning can identify previously unknown malware by analyzing behavior during execution.

Liang et al. (2023) highlighted the importance of automatic feature extraction in improving malware detection results. Their research shows that automatic feature extraction techniques using convolutional neural networks (CNN) can reduce the effort required and improve overall perception. These studies demonstrate the potential of machine learning algorithms to provide powerful and effective solutions to combat cybersecurity threats by improving malware detection and analysis.

Sr. No	Name of Author	Focus	Key Finding	Technique Used
1	Liang et al. (2023)	Automate d feature extraction	Automated feature extraction techniques increase the efficiency of malware detection	Automated Feature Extraction, CNN
2	Salehi et al. (2022)	Adversarial robustness in malware detection	Enhanced resilience of malware detection models against adversarial threats	Adversarial Training, GANs
3	Kumar et al. (2022)	Ensemble learning for malware classification	Ensemble methods enhance accuracy and robustness in malware classification	Ensemble Learning, Random Forest
4	Zeng et al. (2021)	Dynamic analysis using ML	Dynamic analysis methods combined with ML improve detection rates for zero-day malware	Dynamic Analysis, Deep Learning
5	Hou et al. (2021)	Graph-based deep learningfor malware detection	Graph neural networks outperform traditional ML methods in detecting malware	Graph Neural Networks (GNN)
6	Vinayakumar et al. (2019)	Real-world application of ML in malware detection	Large-scale datasets demonstrate the scalability and efficiency of MLmodels	Large-scale datasets, model evaluation
7	Anderson et al. (2018)	Feature extraction in MLmodels	Diverse feature extraction improves malware detection accuracy	Various ML models, feature extraction
8	Raff et al. (2017)	Supervised learning for malware classification	Random forest algorithmsshow high accuracy in malware classification	Random forest, supervised learning

(*continued*)

(*continued*)

Sr. No	Name of Author	Focus	Key Finding	Technique Used
9	Kolosnjaji et al. (2016)	Unsupervised learning for anomaly detection	Deep learning-based anomaly detection uncoversnew malware variants	Deep learning, anomaly detection
10	Papernot et al. (2016)	Robustness against adversarial attacks	ML models are vulnerableto adversarial attacks; defensive strategies are needed	Adversarial attacks, model defence

3 Tools and Technology

In this section, we describe the data and methods used to improve malware detection and analysis using machine learning algorithms. The process includes data collection, feature extraction, model selection, training and analysis. We also include mathematical models that support our approach [1].

Datasets

- Malware samples: We collected several malware samples from various sources, including Virus Total and public databases.
- Benign Samples: Legitimate software samples were gathered to serve as a baseline for comparison.

Software Tools

- **Python**: Used for scripting and implementation.
- **Scikit-learn**: For basic machine learning models.
- **Tensor Flow/Keras**: For deep learning models.
- **Graph Neural Networks Libraries**: For graph-based deep learning (e.g., PyTorch Geometric).

3.1 Model Evaluation

Evaluate the model using metrics such as accuracy, precision, recall, and F1 s core. The evaluation process needs to be validated to ensure robustness.

Proposed Architectures
We offer three integrated models to effectively detect and analyze malware using machine

learning algorithms: static analysis models, evaluation models, and evaluation models. Each company uses different types of malware behavior to provide a comprehensive detection system.

Static analysis methods focus on identifying malware code before it processes it. The model extracts features such as opcode sequences and API call frequencies from binary archives and uses machine learning classifiers to identify malware (Fig. 1).

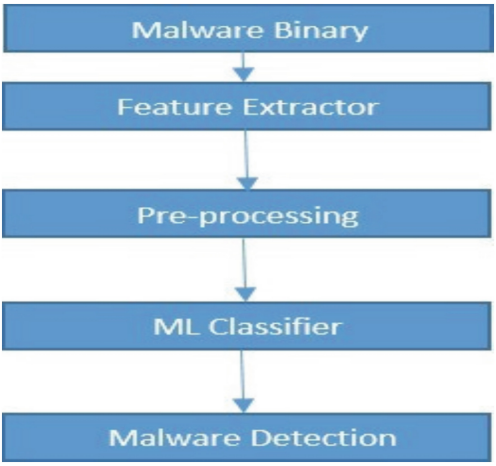


Fig. 1. Static Analysis Model in malware detection using machine learning.

This figure outlines the sequential process of static analysis in malware detection, where machine learning techniques are applied to features extracted directly from the binary file without executing it.

3.2 Integration of Architectures

To achieve optimal malware detection, these three models can be combined in a unified system where static and dynamic features are combined with image analysis, leveraging integrated learning strategies to improve search accuracy and robustness..

1. **Feature Extraction**: Extract both static and dynamic features.
2. **Graph Construction and Feature Extraction**: Construct graphs and extract features.
3. **Machine Learning and Deep Learning Models**: Apply individual models and combine their outputs using an ensemble method.
4. **Decision Fusion**: Fuse the decisions from different models to provide a final malware detection result (Fig. 2).

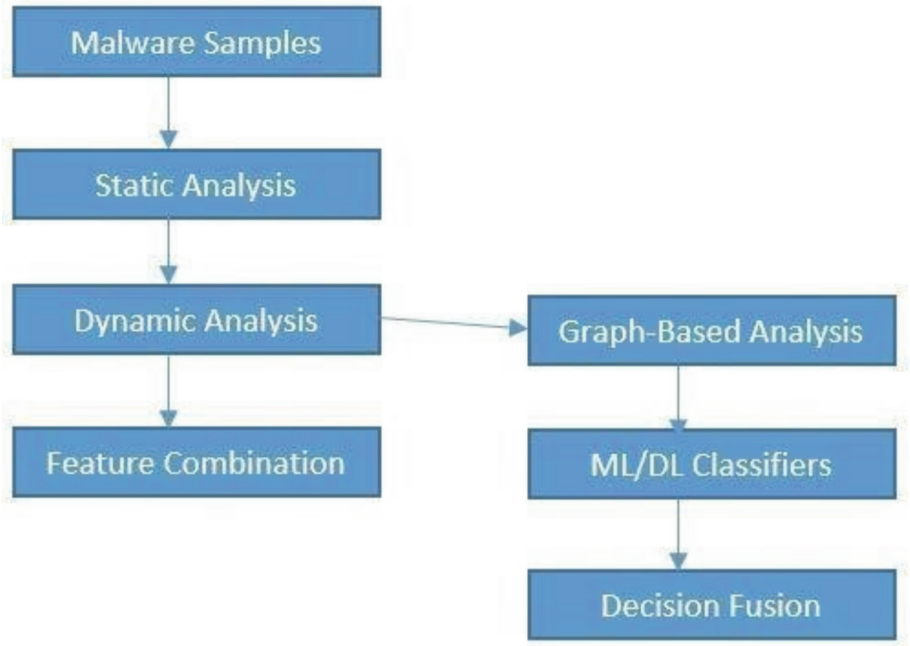


Fig. 2. Integrated Model in Malware detection using ML technique

3.3 Proposed Algorithm for Effective Malware Detection Analysis

Step:-1 Input: Malware samples M and benign samples B.

Step:-2 For Static Analysis Extract static features Fstatic(M) and Fstatic(B) from malware and benign sample relatively.

Step:-3 For Dynamic Analysis Execute malware samples in a controlled environment to capture dynamic features Fdynamic(M).

Step:-4 Combine static and dynamic features: F(M)=Fstatic(M) U Fdynamic(M). Similarly, combine static features for benign samples: F(B)=Fstatic(B)

Step:-5 Train a machine learning model θ using F(M) and F(B).

Step:-6 θ=Train(F(M), F(B))

Step:-7 for new sample S: Extract feature F(s)

Step:- 8 Predict the probability of S being malware using the trained model θ: Prediction = θ(F(S))

Step:- 9 Evaluate the model θ using metrics such as accuracy, precision, recall, and F1-score on a validation set or through cross-validation.

Description of Components

1. **Malware Binary**: Represents the raw binary or executable file of the malware.
2. **Feature Extractor**: Module responsible for extracting static features from the binary file, such as opcode sequences, API call frequencies, or structural metadata.

3. **Pre-processing**: Involves normalization, transformation, and preparation of extracted features into a format suitable for machine learning algorithms.
4. **ML Classifier**: Utilizes machine learning algorithms (e.g., Random Forest, SVM) to classify the extracted features as benign or malicious.
5. **Malware Detection**: Output indicating whether the analysed sample is classified as malware or benign based on the classifier's decision.

4 Results and Discussion

(See Figs. 3, 4 and Tables 1, 2)

Table 1. Training Validation Accuracy

Epoch	Training Accuracy	Validation Accuracy
1	0.85	0.82
5	0.92	0.90
10	0.94	0.93
15	0.95	0.94
20	0.96	0.95

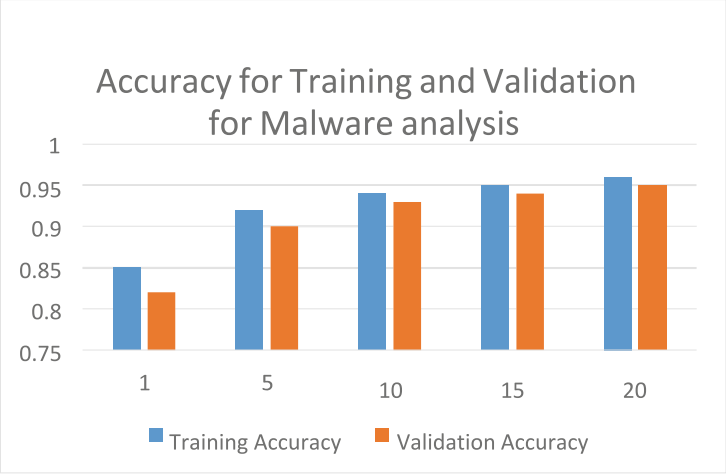


Fig. 3. Accuracy for training and validation analysis

Table 2. Training and Validation loss

Epoch	Training Loss	Validation Loss
1	0.40	0.42
5	0.25	0.27
10	0.20	0.22
15	0.18	0.20
20	0.15	0.18

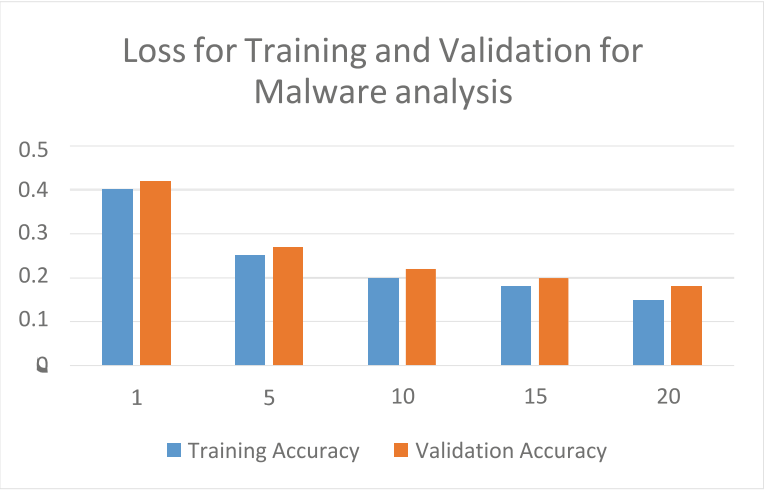


Fig. 4. Loss for training and validation analysis

5 Conclusion

Our research demonstrates the effectiveness of combined static and dynamic analysis in machine learning in detecting malware. Performance measurements and key points of verifying the robustness and reliability of the method. Future work may investigate the integration of additional features and more advanced machine learning techniques, such as ensemble learning and deep neural networks, to improve detecting the latest threats and adapting to changing malware threats.

References

1. Anderson, H.S., Woodbridge, J., Filar, B.: DeepDGA: adversarially-tuned domain generation and detection. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (2018)
2. Bayer, U., Kruegel, C., Kirda, E.: TTAnalyze: a tool for analyzing malware. In: Proceedings of the 15th European Conference on Research in Computer Security, pp. 180-199 (2009)

3. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 506–519 (2016)
4. Saxe, J., Berlin, K.: Deep neural network based malware detection using two-dimensional binary program features. In: Proceedings of the 10th International Conference on Malicious and Unwanted Software (MALWARE), pp. 11–20 (2015)
5. Saxe, J., Berlin, K.: Deep neural network based malware detection using two- dimensional binary program features. In: Proceedings of the 10th International Conference on Malicious and Unwanted Software (MALWARE), pp. 11–20 (2015)
6. Raff, E., Barker, J., Sylvester, J., Brandon, R., Catanzaro, B., Nicholas, C.: Malware detection by eating a whole EXE. In: Workshops at the Thirty-First AAAI Conference on Artificial Intelligence (2017)
7. Kolosnjaji, B., Zarras, A., Webster, G., Eckert, C.: Deep learning for classification of malware system call sequences. In: Australasian Joint Conference on Artificial Intelligence, pp. 137–149. Springer (2016)
8. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, pp. 506–519 (2016)
9. Vinayakumar, R., Soman, K.P., Poornachandran, P.: Evaluating shallow and deep networks for ransomware detection and classification. In: 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 259–265. IEEE (2019)
10. Hou, Y., Jiang, L., Yu, S., Lin, Z.: MalGraph: Malware detection using graph- based deep learning. In: Proceedings of the 2021 ACM Conference on Computer and Communications Security (2021)
11. Salehi, S., Jang-Jaccard, J., Han, Y.: Adversarially robust malware detection using generative adversarial networks. IEEE Trans. Inform. Forensics Security (2022)
12. Kumar, S., Raj, P., Lal, K.: Enhancing malware detection through ensemble learning techniques. J. Cybersecurity Privacy (2022)
13. Zeng, Y., Zheng, J., Chen, J., Han, Q.: Improving zero-day malware detection with dynamic analysis and machine learning. IEEE Trans. Depend. Secure Comput. (2021)



Generative AI Tool in Learning an Introductory Programming: A Research Perspective

M. Sivasakthi(✉) and A. Meenakshi

Department of Computer Science and Applications, FSH, SRM IST, Vadapalani Campus, Chennai, Tamilnadu 600 026, India
sivasakthi.info@gmail.com

Abstract. Since generative artificial intelligence applications have become more diverse, there has been a sharp rise in interest in their application across all societal segments and fields in recent years. Program learning processes are one of these domains. ChatGPT is one of the generative Artificial Intelligence instruments utilized for this. While ChatGPT has some prospective in the process of programming, its application in the process of building programs has grown significantly. The impact of teaching programming to students through ChatGPT on their computational thinking abilities was examined in this research. A study of 160 undergraduate students who took an introductory programming course called "Problem Solving using Python" was carried out. Pretest-posttest control group participation was maintained throughout the study in accordance with the experimental design. The students were divided into experiment and control groups at random. The exploratory gathering understudies gained benefits from ChatGPT during their week after week programming rehearses, while the benchmark group didn't use it. The computational thinking scale was employed to collect research data for the "Problem Solving using Python" course. Results indicated that students in the experimental group exhibited significantly higher levels of computational thinking skills compared to their peers in the control group. Because of this, it is possible to conclude that ChatGPT, a generative AI tool, is beneficial to students and helps them learn and write programming. AI assistance can be most useful while learning and writing code, according to study findings. For academics and educators in this regard, many recommendations were given.

Keywords: ChatGPT · Programming Education · Computational Thinking · Introductory Programming · Experimental Research

1 Introduction

Computer programming is the backbone of computer correlated courses. Learning programming is the key part of students' life in order to attain in their career. The corporate world has become more digitally connected, and students increasingly need programming education to stay competitive in the employment market. This is due in large part to the fast advancements in technology. Students who study programming skills acquire critical thinking, logical reasoning, algorithmic thinking, and problem-solving abilities

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 120–132, 2025.
https://doi.org/10.1007/978-3-031-86296-0_10

that are applicable to all aspects of their lives. These abilities can have a favorable impact on students' growth as individuals, their academic success and performance at work [1]. Besides, learning programming can enhance pupils' inventiveness and capacity for self-expression.

Though there are ample of general tools for learning programming such as BlueJ, Greenfoot, processing students are penetrating more advancement in terms technology related tools connected with Artificial Intelligence for learning programming.One of the main objectives of introductory classes has always been to teach students how to read and comprehend code. LLM-based tools make it very easy to write code, therefore this is a talent that is becoming more and more relevant. Another crucial competence is learning how to create persuasive hints that inspire LLMs to write code. Although many common introductory questions may be answered by LLMs using exact quotes from textbooks or exam issue statements, recent research has demonstrated [2] and [3].

Students can improve the readability and simplicity of their programming code by using visual programming tools. In contemporary discourse, the concept of developing artificial intelligence has emerged as an important topic in various fields. This trend is particularly evident with the growing integration of AI language models such as ChatGPT into the everyday experiences of individuals. The field of software development and programming is one of these.

The purpose of programming education platforms and tools is to improve students' enjoyment and effectiveness in their learning journey. The purpose of these resources is to facilitate understanding of programming principles for students [4].This is the reason why a lot of people who begin studying programming may give up before they reach the goal of learning programming. The need for assistance and direction when developing the program, the challenge of troubleshooting the mistakes in the program they have developed, etc., are some of the causes of this. These days, people can get quick, largely accurate responses to programming-related questions via generative AI tools like ChatGPT. It's almost like it can give them advice and assistance. This investigation seeks to explore the benefits of incorporating ChatGPT into the educational journey of students in the field of programming. According to a review of the literature, no study has looked into how students feel ChatGPT helps with programming education.

Hwang proposed an AIED model to exemplify Artificial Intelligence in Education in 2020, based on which we categorize the Generative AI in the field of education hooked on variouskinds, "teacher, student, learning peer, domain expert, administrator, and learning tool" [5].

Teacher/Tutor: offering more resources, summarizing the topic being learned, exhibiting, etc.
Student/Tutee: Learning from peers and completing assignments as requested by peers
Learning peer/Partner: Participating in group as a member for cooperative learning
Domain expert: Giving counsel or responding to specific problems or inquiries
Administrator: Providing an overview of the research to help with decision-making
Learning tool: Assisting students in gathering and analyzing data, collaborating with students, and allowing students to concentrate on important goals by sharing their workloads.

Numerous academicians have highlighted the benefits of GAI-based learning, such as an increase in students' abilities for creativity, critical analysis, and problem-solving. Several motivations drive the research report for the introductory programming course. First, it was questioned how students' programming abilities would be impacted by ChatGPT's benefits when it came to producing code. One of the researcher taught the students a variety of programming languages, and his area of expertise is the pedagogical elements of computer programming, which is another rationale for this study.

ChatGPT is best at helping with specific coding jobs or procedures, not from scratch program development. Use ChatGPT to locate and select the best coding libraries for a certain task, then have an interactive conversation to reduce the number of possibilities. ChatGPT can help students grasp programming ideas better and save time by offering rapid answers to basic queries regarding syntax, algorithms, and data structures. Additionally, Chat-GPT can help students with more difficult code difficulties by offering examples, making suggestions for possible solutions, or succinctly and clearly outlining coding principles.One can use Chat-GPT to enhance your coding skills by doing the following.

Step 1: Pose questions of thought
Step 2: Look for code samples
Step 3: Go over outstanding practices
Step 4: Make code reviews requests
Step 5: Complete the code tasks
Step 6: Help with debugging
Step 7: Test and refine your approach.

OpenAI'sChatGPT, a large language model, was used to teaching and learning programming in this study. The GPT design fills in as the establishment for ChatGPT, which was prepared utilizing a sizable text dataset from the web. The ChatGPT can be improved for specific assignments including question addressing, language interpretation, and text outline. It will likely deliver content that seems human. 2018 saw the release of GPT-1. A sizable linguistic dataset is used to train this model. In 2019, ChatGPT's second edition, GPT-2, was made available. 2020 saw the release of GPT-3. After GPT3.5 the GPT-4 version has been in use in recent days. GPT-4 is said to be more collaborative and innovative than earlier iterations [6].

In this study, we examine to identify a significant difference in performance on the following skills concerned with computational thinking between the students in the control and experimental groups.

- Algorithm
- Creativity
- Critical
- Cooperativity
- Problem solving

2 Context and Review of Existing Literature

Interest in General Artificial Intelligence (GAI) in the field of computing education has experienced a significant surge in recent times. The ACM Digital Library now hosts a collection of 37 papers organized by ACM SIGCSE with ALM, showing the growing discourse on this topic. Notably, a paper by James Brather and colleagues from the ITiCSE 2023 Task Force examines the transformative impact of generative AI on computing education. Also, recent research by Fred Becker and team sheds light on the challenges and possibilities associated with code generation through AI. The integration of GAI tools into introductory programming is explored in the insightful book "Learn AI-Assisted Python Programming with GitHub Copilot and ChatGBT" by Leo Porter and Daniel Zingaro. This wave of innovation from GAI promises to revolutionize the landscape of computer education beyond conventional boundaries [9].

AI-powered tools that offer recommendations, mistake detection, and automatic code production can help students learn how to code. Students' time and effort in finishing their programming tasks can be decreased by helping them write code that is efficient and valid. AI-driven environments and tools possess the capacity to enhance students' motivation and engagement through interactive features that offer individualized guidance and support to learners of programming. YilmazKaraoglan and Yilmaz R, research article, conclude that the students are satisfied with smart mooc courses [10]. Conversely, ChatGPT interprets user requests in plain language and accurately responds to them by utilizing machine learning and natural language processing technology. ChatGPT is designed to comprehend user requirements and adapt accordingly, without utilizing programming language syntax or notions. For this reason, ChatGPT provides instructors and students with an alternative method of teaching programming [11].

Our evaluations of the literature indicate that researchers have looked into ChatGPT's potential for computer programming instruction and learning in great detail [12–14]. Code snippet recommendations, syntax checks, and debugging assistance are all provided by automatically created AI content to developers. Literature does in fact indicate that ChatGPT is a useful technique for producing snippets of computer code. As demonstrated in [15] and [16], it is also useful for carrying out thorough reviews and surveys of the literature, which aids scientists in producing material for research proposal recommendations. [17] and [18] GitHub Copilot is the best AI-based assistant, though, if you're looking for one that's especially designed for programming [19].

The publications referenced in this passage demonstrate that AI-generated material may quicken the progress of process improvements, reduce errors in coding and enhance the entire software engineers' efficiency. Also, ChatGPT provides an opportunity for individuals with minimal or no programming knowledge to engage with principles of computer science and coding. This innovative tool has the potential to improve software development processes and increase the efficiency of engineers. As mentioned earlier, ChatGPT enables the creation of system code and software prototypes, enabling software engineers to streamline their workflow and spend more time designing complex tasks and tests. Additionally, engineers have the ability to obtain justifications, suggestions, also utilize genuine online simulations that improve practical learning experiences.[21]. An excellent illustration is the utilization of ChatGPT, which empowers developers to effortlessly generate code at a reasonably satisfactory standard, complete any omitted

sections and/or modify it, or even create it in diverse programming languages [22]. Furthermore, ChatGPT possesses a significant feature of providing explanations and ideas, which can significantly reduce the errors in coding and increase the totalvalue of the codewritten in programming [20].

ChatGPT facilitates students' progression to more complex programming levels. In programming, students can move to more complex levels. Studies reveal that students' interest, computational thinking, and programming self-efficacy all rise when using generative AI tools like ChatGPT. The purpose of programming education platforms and tools is to improve students' enjoyment and effectiveness in their learning journey. The purpose of these resources is to facilitate understanding of programming principles for students [11]. Eason Chen et alconduct a study on "ChatGPT-powered programming tool" and claims that by offering advantages like debugging and code explanations, The purpose of programming education platforms and tools is to improve students' enjoyment and effectiveness in their learning journey[23].

A number of publications have analyzed the effectiveness of ChatGPT for educational purposes, with Enkelejda Gaznesi et al highlighting the potential benefits and barriers of using large language samples in education [24]. Additionally, a review by C. Kin Lo examines the educational impact of ChatGPT [25]. Given the limited research in this area, it is critical to examine the unique effects of ChatGPT-assisted instruction on students' understanding of programming processes and their academic outcomes. In the absence of existing studies, the impact of ChatGPT on programming education remains unexplored.

3 Methodology

3.1 AIED Framework

The emerging discipline of Artificial Intelligence in Education (AIED) uses cutting-edge technologies to revolutionize the way students create learning materials and educators create teaching strategies. To date, meanwhile, the majority of research on AIED has examined the technology side of things, making it impossible to fully comprehend the intricate roles that AI plays in education. In 2022, Weiqi Xu and Fan Ouyang presented a framework for Artificial Intelligence in Education which is shown in Fig. 1. The framework consist three kinds of relationships in learning by using AI. Immersion of AI in the instructor-student relationship for instructional process, AI implicates in the student-student relationship for the intension of collaboration. AI embroilsin the student-self relationship for learning process [26]. ChatGPT's response to swap the first and last element of a list has shown in Fig. 2.

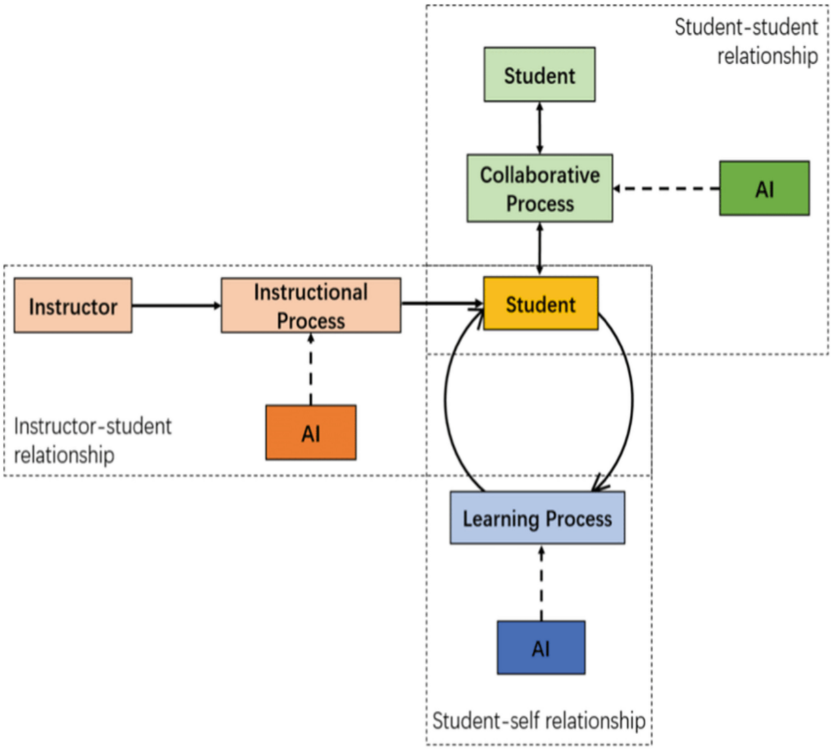


Fig. 1. AIED framework

3.2 Experimental Framework

The experimental group received the specialized intervention, while the control group did not participate in the treatment. In the scope of this study, the experimental team had the privilege of using the ChatGPT tool for their preliminary programming work, specifically in the course "Problem Solving Using Python" at a prestigious university in Chennai, Tamil Nadu, India. Over a six-week period, students in the experimental group incorporated the intervention efficiently, leading to a significant improvement in their understanding. As a result, the impact of the intervention on students' perceptions was evident. Both groups underwent a post-test at the end of the study to assess their progress post-intervention.

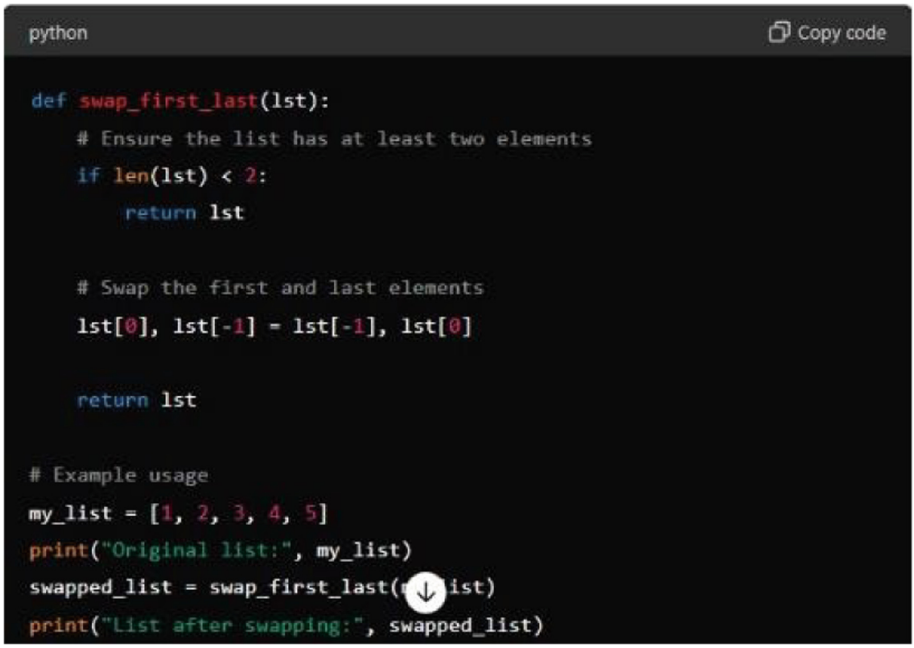


Fig. 2. To present ChatGPT's elegant solution to the desired streamlined program.

To assess the impact of the intervention, the researchers compared the changes in scores between the experimental and control groups before and after the intervention. Figure 3 neatly visualizes the research methodology used in this study.

3.3 Participants and Process

Undergraduate computer science students at Chennai's affiliated institution of the University of Madras, Chengalpattu, Thiruvallur, and the Kanchepuram district of Tamilnadu, India, were the subjects of the study. In this research study, two hundred students enrolled in the "Problem Solving using Python" course took part. Nonetheless, a portion of the student body did not show up for the lesson during the procedure, and some did not complete the pretest and posttest. Due to their active involvement in this program, a total of 160 students have successfully cleared both the pre-examinations and postgraduate examinations.

Within the experimental group, the experimental group consisted of 78 individuals, whereas the comparison group consisted of 82 participants. Among the students involved in the research, 65 were male and 95 were female. Students completed C programming assignments as part of the study's scope. The research participants' prior training in "Problem Solving using Python" was nonexistent. The pupils in the control group didn't make use of ChatGPT. Everything else is the same, including the lab assignments, instructional strategies, and the teacher for both groups.

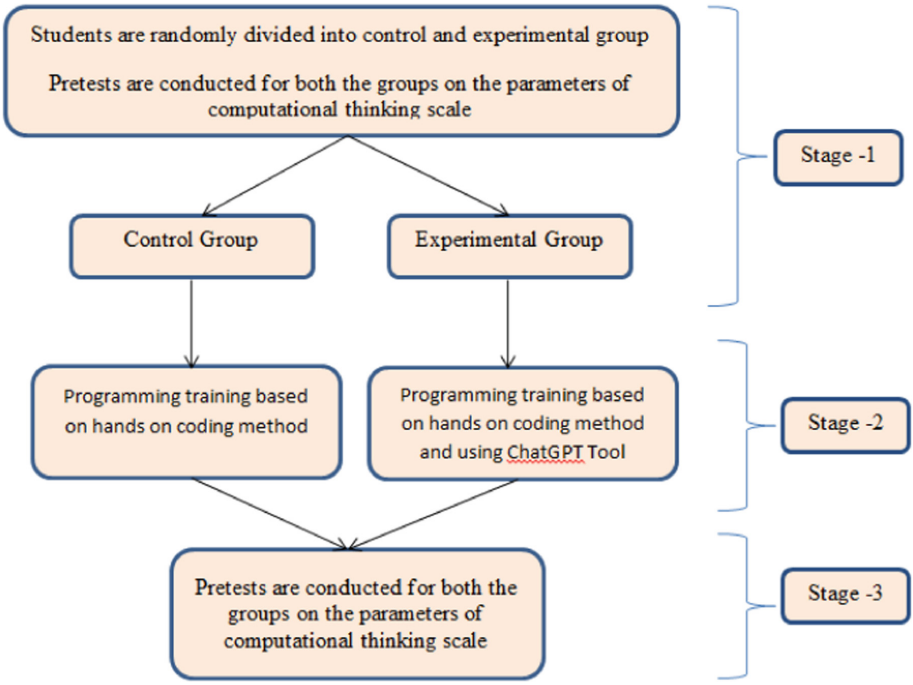


Fig. 3. Research process

3.4 Data Collection

The computational thinking scale is the scale that was utilized as a pretest and posttest in the study. Korkmaz et al. created the computational thinking scale in 2017 which has been adopted for this study. Creativity, Problem-solving, cooperativity, algorithmic thinking, and critical thinking are the five sub-factors that compose the scale's twenty-eight items.Superior scores are indicative of a more advanced level of proficiency in computational reasoning. Utilizing a five-point Likert scale, the metric is assessed. The Cronbach alpha reliability values for algorithmic thinking, inventiveness, critical thinking, cooperativity, and problem-solving were found to be 0.85, 0.88, 0.87, and 0.75 respectively when the scale's reliability was recalculated utilizing the information gathered in this investigation. The entire scale's overall dependability value was determined to be 0.84.

3.5 Data Analysis

It was found during the analysis phase that the data displayed a normal distribution. Next, the students' pretest results were manipulated, and it was determined if the posttest results differed from the pretest results in a meaningful way. Using SPSS, the ANCOVA test was utilized to ascertain and attempt to explain the association between the independent and dependent variables.

The study groups' descriptive statistics about their computational thinking skills are displayed in Table 1. Additionally, it demonstrates that the students' pretest scores for computational thinking abilities are comparable. A neat and reliable analysis was conducted by analysis of covariance to assess the development of computational thinking skills. The results of the ANCOVA analysis are neatly presented in Table 2, which shows significant improvements in various areas such as creativity, algorithmic thinking, collaboration, critical thinking and problem solving based on the pre-test scores. Notably, there were substantial increases in scores post-intervention across all domains. For computational thinking skills, the pretest mean was 8803.19 and significantly rose to 6754.98 post-intervention ($F = 91.39$, $p < 0.001$, $\eta 2 = 0.3087$). Similarly, creativity scores increased from 795.97 to 720.52 ($F = 129.50$, $p < 0.001$, $\eta 2 = 0.4274$). Algorithmic thinking also saw a notable enhancement, with scores rising from 410.55 to 314.21 ($F = 39.99$, $p = 0.002$, $\eta 2 = 0.1631$). Cooperativity, critical thinking, and problem-solving exhibited comparable trends, with all showing statistically significant improvements post-intervention ($p < 0.05$). These findings underscore the efficacy of the intervention in fostering diverse cognitive skills among participants.

Table 1. Statistics describing the computational thinking skills

Parameters	Tests	Groups	Mean	S.D
Computational Thinking Skills	Pretest	Experimental	410.034	39.235
		Control	401.376	62.37
	Posttest	Experimental	468.901	29.19
		Control	416.657	53.62
Creativity	Pretest	Experimental	122.951	11.235
		Control	119.88	21.49
	Posttest	Experimental	137.492	6.86
		Control	119.695	15.05
Algorithmic Thinking	Pretest	Experimental	76.701	12.355
		Control	74	12.635
	Posttest	Experimental	92.13	9.975
		Control	80.66	14.7
Cooperativity	Pretest	Experimental	59.385	12.11
		Control	59.2	10.36
	Posttest	Experimental	66.933	6.615
		Control	61.309	10.395
Critical thinking	Pretest	Experimental	70.818	10.92
		Control	70.929	10.325
	Posttest	Experimental	79.735	9.94
		Control	68.376	19.005

(*continued*)

Table 1. (*continued*)

Parameters	Tests	Groups	Mean	S.D
Problem solving	Pretest	Experimental	80.216	8.47
		Control	78.662	22.015
	Posttest	Experimental	92.5	10.465
		Control	82.362	14.315

Table 2. Covariance analysis results pertaining to computational thinking skills

Scales	Pre-post tests	Some of Square	Degrees of Freedom	M.S	F Value	P Value	Partial eta squared
Computational Thinking Skills	Pretest	8803.19438	1	8803.19438	91.38879	0.000	
	Group	6754.97785	1	6754.97785	70.1256	0.000	0.3087
	Error	15123.31524	157	96.32684866			
	Total	30681.48747	159				
Creativity	Pretest	795.96529	1	795.96529	129.4961	0.000	
	Group	720.51629	1	720.51629	117.2213	0.000	0.4274
	Error	965.02159	157	6.146634331			
	Total	2481.50317	159				
Algorithmic thinking	Pretest	410.55447	1	410.55447	39.98625	0.002	
	Group	314.2144	1	314.2144	30.60314	0.007	0.1631
	Error	1611.98052	157	10.26739185			
	Total	2336.74939	159				
Cooperativity	Pretest	136.30999	1	136.30999	26.26428	0.011	
	Group	92.85281	1	92.85281	17.89093	0.034	0.1022
	Error	814.82032	157	5.189938344			
	Total	1045.61484	159				
Critical thinking	Pretest	514.50803	1	514.50803	33.10624	0.005	
	Group	276.81841	1	276.81841	17.812	0.035	0.1018
	Error	2439.95568	157	15.54111898			
	Total	3231.28212	159				

(*continued*)

Table 2. (*continued*)

Scales	Pre-post tests	Some of Square	Degrees of Freedom	M.S	F Value	P Value	Partial eta squared
Problem solving	Pretest	482.73281	1	482.73281	49.66525	0.001	
	Group	270.63809	1	270.63809	27.8442	0.009	0.1506
	Error	1525.99754	157	9.719729554			
	Total	2279.36844	159				

4 Result and Discussion

Hence, through the utilization of ChatGPT, the learner can obtain the desired code fragments' responses by presenting the most relevant inquiries. A difficult aspect of programming instruction is that the students in the control group devoted more time to problem-solving than to code development, integration, or debugging. Our findings are supported by an earlier study by Huang and Qiao, which discovered that a notable disparity in computational thinking abilities was observed between students in the experimental group underwent artificial intelligence training using a STEAM model, and those in the control group, which did not partake in such instruction. The usage of voice assistants in the course significantly improved students' computational thinking abilities, according to our results and earlier research by Hsu et al. Additionally, our findings, in conjunction with García et al., revealed that teaching students in machine learning and artificial intelligence was successful in enhancing their computational thinking abilities. Overall analysis of the data indicates that deploying AI-supported tools and offering students AI training were successful in raising students' computational thinking abilities. It may be argued that integrating ChatGPT into programming instruction helped students become more proficient in computational thinking.

5 Conclusion and Recommendations

The impact of ChatGPT on computational thinking abilities for college students taking programming courses was investigated in this study. The experimental design guided the conduct of the study. The study's findings showed that using ChatGPT to teach programming to students significantly and statistically improved their computational thinking abilities. It was determined that the application of ChatGPT substantially modified the sub-dimensions of the scale, including creative thinking, problem solving, algorithmic thinking, cooperative thinking, and critical thinking. Students must be taught timely writing techniques if they are to get the greatest benefits from AI environments and technologies like ChatGPT. Students will be able to use ChatGPT and other technologies effectively and efficiently if they can write good prompts. Teachers should become proficient in AI literacy when incorporating ChatGPT and other similar tools into their lectures, especially so that students can develop prompt writing abilities. When it's thought that developing students' critical thinking abilities is essential for them to reap

the full benefits of instruments like artificial intelligence, educators can use metacognitive tactics. Setting a cognitive goal allows students to reflect and evaluate their learning processes in a subtle and purposeful way. Students can better comprehend, and manage their own learning processes through the implementation of teaching strategies. Students can challenge their own ideas and take charge of their own learning processes when teachers pose questions.

References

1. Kocak, O., Coban, M., Aydin, A., Cakmak, N.: The mediating role of critical thinking and cooperativity in the 21st century skills of higher education students. Think. Skills Creat. **42**, 100967 (2021)
2. Finnie-Ansley, J., Denny, P., Becker, B. A., Luxton-Reilly, A., Prather, J.: The robots are coming: Exploring the implications of openai codex on introductory programming. In: Proceedings of the 24th Australasian Computing Education Conference, pp. 10–19 (2022)
3. Finnie-Ansley, J., Denny, P., Luxton-Reilly, A., Santos, E.A., Prather, J., Becker, B.A.: My ai wants to know if this will be on the exam: Testing openai's codex on cs2 programming exercises. In: Proceedings of the 25th Australasian Computing Education Conference, pp. 97–104 (2023)
4. Beckem, J.M., Watkins, M.: Bringing life to learning: immersive experiential learning simulations for online and blended courses. J. Asynchron. Learn. Networks **16**(5), 61–70 (2012)
5. Hwang, G.J., Xie, H., Wah, B.W., Gašević, D.: Vision, challenges, roles and research issues of Artificial Intelligence in Education. Comput. Educ. Artific. Intell. **1**, 100001 (2020)
6. OpenAI. ChatGPT (2024). https://chat.openai.com/
7. Prather, J, et al.: The robots are here: Navigating the generative AI revolution in computing education. In: Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education, pp. 108–159 (2023)
8. Becker, B.A., Denny, P., Finnie-Ansley, J., Luxton-Reilly, A., Prather, J., Santos, E.A.: Programming is hard-or at least it used to be: Educational opportunities and challenges of ai code generation. In: Proceedings of the 54th ACM Technical Symposium on Computer Science Education, vol. 1, pp. 500–506 (2023)
9. Porter, L.: Learn AI-assisted Python Programming: With GitHub Copilot and ChatGPT. Simon and Schuste (2024)r
10. Yılmaz, R., Yılmaz, F.G.K.: Examining student satisfaction with the use of smart mooc. International İstanbul Scientific Research Congress (2022)
11. Yilmaz, R., Yilmaz, F.G.K.: The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. Comput. Educ. Artific. Intell. **4**, 100147 (2023)
12. Biswas, S.: Role of ChatGPT in computer programming: ChatGPT in computer programming. Mesopotamian J. Comput. Sci. 8–16 (2023)
13. Chen, E., Huang, R., Chen, H.-S., Tseng, Y.-H., Li, L.-Y.: GPTutor: a ChatGPT-powered programming tool for code explanation. arXiv preprint arXiv:2305.01863 (2023)
14. Surameery, N.S., Shakor, M.Y.: Use chat gpt to solve programming bugs. Int. J. Inform. Technol. Comput. Eng. ISSN: 2455-5290 3 **01**, 17–22 (2023)
15. Sallam, M.: The utility of ChatGPT as an example of large language models in healthcare education, research and practice: systematic review on the future perspectives and potential limitations. Healthcare **2023**, 11 (2023)

16. Devlin, J., Chang, M., Lee, K., et al.: Bert: pre-training of deep bidirectional transformers for language understanding (2018).arXiv:1810.04805
17. Liu, Y., Ott, M., Goyal, N., et al.: Roberta: a robustly optimized bert pretraining approach (2019). arXiv:1907.11692
18. Li, W.: Iaso: Enhancing syntax error correction using deep learning techniques. Research-Square (2023)
19. Prather, J., Denny, P., Leinonen, J., Becker, B.A., Antonio Santos, E.J.: It's weird that it knows what I want: usability and interactions with copilot for novice programmers. ACM Trans. Comput.-Hum. Interact. **31**(1), 1–31 (2023). https://doi.org/10.1145/3617367
20. Haque, M., Li, S.: The potential use of ChatGPT for debugging and bug fixing. EAI Endorsed Trans. AI Robot. **2**(2) (2023)
21. Qadir, J.: Engineering education in the Era of ChatGPT: promise and pitfalls of generative AI for education. In: IEEE Global Engineering Education Conference (EDUCON), pp. 1–9. Kuwait (2023)
22. Kreitmeir, D.H., Raschky, P.A.: The unintended consequences of censoring digital technology – evidence from Italy's ChatGPT Ban.2023. SocArXiv
23. Chen, E., Huang, R., Chen, H.S., Tseng, Y.H., Li, L.Y.: GPTutor: a ChatGPT-powered programming tool for code explanation. In: International Conference on Artificial Intelligence in Education, pp. 321–327. Springer Nature Switzerland, Cham (2023)
24. Kasneci, E., et al.: ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individ. Differ. **103**, 102274 (2023)
25. Lo, C.K.: What is the impact of ChatGPT on education? A rapid review of the literature. Educ. Sci. **13**(4), 410 (2023)
26. Xu, W., Fan, O.: A systematic review of AI role in the educational system based on a proposed conceptual framework. Educ. Inform. Technol. Official J. IFIP Techn. Committee Educ Springer Link **27**(4) (2022). https://doi.org/10.1007/s10639-021-10774-y

Evaluating the Effectiveness of Machine Learning Algorithms in Predicting Plant Diseases: A Comparative Analysis

Ashish K. Patil(✉) and Deepak Kumar Yadav

SAGE University, Indore, M.P, India
ashish.patil1389@gmail.com

Abstract. In the recent years there have loss in the production of the agricultural yields due to the disease of the plants. Latest emerging technologies like machine learning, deep learning and Internet of things address the early prediction of these plant disease and minimize the crop loss. Deep learning technology extract the complex features from the image dataset of the plants and train the model. In this paper we perform the analysis of the literature review for the plant disease. Convolutional Neural Network (CNN), Support vector machine (SVM) and various deep learning models were used for the classification of the healthy and diseased plant images. Latest deep learning algorithm like YOLO (You Only Look Once) will be used for the object detection and imprve the accuracy of the model.

Keywords: Machine learning · deep learning · Internet of things · SVM · CNN · YOLO · Plant disease

1 Introduction

Latest technologies such as Machine learning (ML), deep learning (DL), Edge Computing and Internet of everything improved the agricultural practices and enhance agriculture production. This helps to achieve food security for human beings. Using latest technologies we can predict the disease or illness of the plant and notify the farmers to take accurate actions according to illness of the plant. This helps to prevent loss of the agriculture production and helps the farmers to get more yields. Globally farmers are more depends on the nature and facing the challenges such as uncertainty of the rainfalls, variation in the range of the temperature, soil pollution, attack of the insects, virus, bacteria, fungus on the plant. This ultimately result in very high cost of production and farmers get very less production and leading to debt trap. In this research paper our aim is to early prediction of the plant disease using images of the leaf and use of latest technologies such as machine learning, deep learning. Alert the farmers for avoid possible loss in the future and recommended that farmers can take appropriate actions.

Crop disease is classified as Bacterial disease, viral disease and fungal disease. Viral diseases have severely affected to the fruits and vegetable crops and that result to the severely affected to the health and productivity of the crop. Fungal diseases were severely

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 133–143, 2025.
https://doi.org/10.1007/978-3-031-86296-0_11

affect the leaves, stems, roots, and fruits of the plant, resulting into loss of agriculture production. Tumors, necrosis, soft rot, and vascular wilt are a few common types of bacterial diseases and this affects severely to the plants.

For the plant disease prediction using machine learning there having some issues and challenges. Most important issue is accurate data collection for a particular crop. The data collection in the form of the images of the leaf and steam of the crop. The images quality of the leaf of the plant must high and remove the noisy data from the dataset is required to train the model accurately [20]. After training the model accurately segmentation and classification of the plant images accurately is also major challenge. Numerous illnesses, including bacterial, fungal, and viral ones, can affect different types of plants, making disease detection a challenging task.

For more accurate prediction and model generalization more extensive dataset based on geographical location, environmental condition and variety of the plants are required.

1.1 Plant Disease Impact on Food Security

All the animals and plants directly or indirectly depends on the plant for the food. Plant disease causes significant impact on the food security worldwide. As a result of the decline of the yield loss world faces severe food shortages. This directly effect on nutrition security of the world.

In this section we discussed about introduction of the plant disease prediction and challenges face for the same. In the next section we see detail about deep learning technologies used for the plant disease.

2 Deep Learning for Plant Disease Prediction

Traditional approach of the machine learning depends on manual intervantion by the expert. This will be depend on small number of the samples for a single crop. In contrast of the traditional approach, deep learning model gives high accuracy and performance. Deep learning algorithm extact significant complex feature from the image dataset and get high accuracy [4].

DL algorithms have demonstrated their utility in plant disease diagnosis and prediction. These algorithms accurately detect and identify plant diseases by utilizing neural network power to Easily recognize characteristics in pictures.

CNN (Convolutional Neural Network) is popular algorithm used in the deep learning. In this algorithm it will extract the features from the images and use this images for the classification or detect the objects. Convolutional, pooling, and fully linked layers are among the layers that make up CNN. CNN's job is to reduce the number of images while maintaining the attributes necessary for precise prediction.

Feature extraction of achitecture of the CNN is as shown in Fig. 1. Convolution layer applies set of the filters to the input images and perform the dot product between filter weight and corrosponding dot product. The pooling layer is reduce the dimension of the convolution layer using max pooling or aveage pooling techniques which is best suitable. The pooling layer progressively shrinks the spatial dimensions of every matrix in a feature map, hence decreasing the number of parameters and computation required

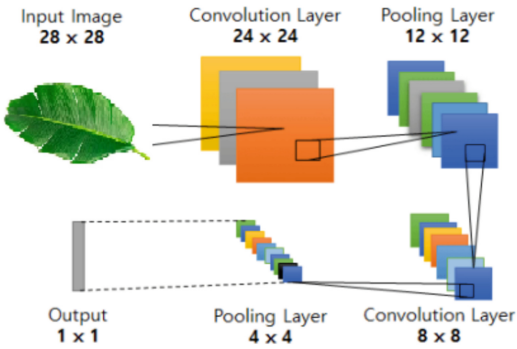


Fig. 1. CNN Feature extraction

by the network. Once more, features go through a number of pooling and convolutional layers before becoming fully linked layers, which are utilized to make precise predictions.

Convolution ⇨Max pooling ⇨Convolution ⇨Max pooling in this process resultant image is smaller than the original image.

For object detection, the YOLO technique is used. We can detect the plant leaf as object for plant disease prediction using YOLO algorithm. This technique is used to recognize objects in photos and videos in real time. Every image in the YOLO algorithm is split into N grids, each with equal dimensions of S × S.

In this each N grid, algorithm detects and localize the objects. The objects within each cell's bounding boxes are predicted using YOLO. These bounding boxes provide the center coordinates (x, y), width and height (w, h), and a confidence score that represents the likelihood that an object is there. The class prediction is then carried out using YOLO. YOlO predicts not only the bounding boxes but also the class probabilities for every box. A probability for each bounding box is assigned to each class label.



Fig. 2. Object detection using YOLO algorithm

Bounding boxes are the outlines that draw attention to specific objects in images, as seen in Fig. 2 above. The following characteristics make up each bounding box: [C, bx, by, bh, bw, C1, C2, C3].

Here C is represent the likelihood of the class object means object is present or not. Bounding box center is represented by bx and by. Height and width of the image

is represented by bh and bw. C1, C2 and C3 is represents the number of the objects present in the bounding box. For the measurement of localization accuracy and calculate localization error in object YOLO calculates intersection over union (IoU). For getting IoU first take intersecting area between bounding boxes for a particular prediction and train bounding box of the same area.

IoU = Area of the overlap / Area of Union

IoU guarantees that the object's real bounding box and its expected bounding box match. The final detection will be made up of the distinct bounding boxes that precisely suit the objects. The next step in the YOLO algorithm is elimination of the duplicate detection using non-maximum suppression. In this phase YOLO looking at the probability score for each box and taking box which have largest probability score. The YOLO calculates IoU values for each box. If IoU is greater than 0.5 the we consider it is positive prediction else it is negative prediction.

The steps for the execution of the YOLO is as follows.

Input image division in to grid cells ⇨Bounding box prediction ⇨ass prediction ⇨eliminates the duplicate detections using non-maximum suppression ⇨final output includes bounding boxes, labels and confidence score.

Farmers must visually inspect plants in order to identify plant diseases using traditional methods, It is labor-intensive and prone to mistakes made by people.

In addition, early illness symptoms are frequently undetectable and challenging to identify. Computer vision systems are able to precisely detect illness indicators in plant photos that may not be evident to the human eye. Through the use of DL algorithms, these systems are able to identify patterns and provide highly accurate disease predictions. But this deep learning techniques have certain limitations like lack of high quality dataset, limited scalability and interpretation of the deep learning model.

In the above section we discuss about deep learning techniques CNN, YOLO algorithm for plant disease prediction. The literature review for machine learning-based plant disease prediction will be covered in the next section.

3 Literature Survey

This section covers a review of the literature on the efficacy of DL and ML methods for the early diagnosis of plant diseases.We have conducted an extensive literature review to investigate, evaluate, and analyze the effects of Machine Learning on agriculture sector. To gather relevant information, we obtained research articles from databases such as Scopus index journals.

Using ML and deep DL techniques, I. Ahmed and P. K. Yadav (2023) forecast plant illnesses in rice, apples, beans, potatoes, and tomatoes.

The author makes use of the 17 disease plant village dataset [1]. In this paper, prediction models were created using SVM, CNN, and GLCM. The suggested method yields an overall accuracy of 98%, 98%, and 96% for apples, rice trees, and 94%, 95%, and 97% for potatoes, beans, and tomato trees, respectively [1]. A machine learning model was jointly created by Shifat E. Arman, Md. Abdullahil Baki Bhuiyan, and others to predict the diseases Sigatoka, Cordana, and Pestalotiopsis that affect banana plants [2]. The Banana Leaf Spot Disease dataset is used in this paper. The author of this research

uses machine learning approaches to classify diseases. Based on GLCM texture features, Chimango Nyasulu, Awa Diattara, Assitan Traore, and other authors' study used Artificial Neural Networks (ANN), KNN, RF, and SVM as classifiers for the classification of tomato fungal leaf diseases. The study's findings showed that GLCM texture characteristics and ML might be used to classify tomato fungal infections [3]. The GLCM is a statistical method for analysing texture in photos The texture of the image is described and the spatial relationship between pixels is taken into consideration by counting the instances in which pairs of pixels with specific values and at a specific geographic location occur in the image.

When comparing the performance and application of ML and DL techniques in the identification of plant diseases, C. Jackulin and S. Murugavalli demonstrate that DL models outperform ML models. The authors discuss using SVM and decision tree (DT) classifiers for classification and prediction tasks [5]. A range of pre-processing techniques, such as RGB conversion, histogram equalization, and contour tracing, are used by S. S. Harakannanavar, J. M. Rudagi, and other authors to improve tomato leaf sample quality and extract useful attributes. Features are extracted from the leaf samples using a variety of descriptors, such as the Principal Component Analysis, Discrete Wavelet Transform, and GLCM.

ML techniques like SVM, CNN, and KNN produce high accuracy rates of 88% to 99.6% when applied to tomato disordered samples [6]. PlantVillage is a dataset used by Imtiaz Ahmed, Gousia Habib, and other researchers. Pictures of leaves for different crop-disease combinations are included. Another component of the data is the pre-processing of images depicting plant diseases. Textural data is extracted using the GLCM method. Disease detection and classification involves the use of SVM, CNN, and KNN.

An evolutionary algorithm, the BOA is named after the way butterflies eat and mate. It has been used in a variety of fields, such as the use of ML to predict plant disease. Ramanathan G and Sevuga Pandian are uses BOA and SVM can accurately predict plant leaf diseases, which can aid in early detection and treatment [8]. The combination of BOA and SVM has the potential to enhance the quality of solutions and improve system performance in addressing engineering optimization problems. The dataset that P. Karthikeyan, Arvind, G. Kumar, and Ms. Saranya imported from Kaggle shows three distinct plant species.

These are apple, tomato, and corn. Apple and corn each have four classes, whereas tomatoes have eleven. Based on photos of afflicted plant leaves, the research proposes the use of a machine learning method called the EfficientNet B3 model for plant disease prediction. The algorithm achieves high precision in the test dataset, with almost 98% accuracy across all classes and varieties of plants, and reliably detects diseases from the photos of diseased plant leaves [10]. Laxmanrao, N. Supriya and A. Arulmurugan were used dataset in this paper was collected from Kaggle and is referred to as the plantvillage dataset. Images of fifteen distinct plant leaf types, such as tomato, potato, and pepper bell are included.The datasets for disease diagnosis were subjected to three different transfer learning techniques: VGG16, RESNET50, and Inception [11]. RESNET50 achieved a 98.7% accuracy for the Tomato dataset, VGG16 achieved a 98.6% accuracy for the Potato dataset, and Inception scored a 99% accuracy for the Pepper Bell dataset.

T. R. Ganesh Babu, S. Priya and other authors propose A deep CNN model and trained using a dataset of plant leaf images to detect plant leaf diseases. The project also mentioned the use of k-means clustering, a popular method for cluster analysis, for disease detection and segmentation [12]. It highlights the importance of smart agriculture and the prevalence of fungal, viral, and bacterial species as the causes of plant diseases, and proposes the use of deep CNN and data augmentation methods to improve disease detection and classification. EnC-SVMWEL (Ensemble Convolutional-SVM Weighted Average Ensemble Learning), an ensemble-based method, has been proposed by Uvarani Vignesh and Bala Subramanian Chokkalingam to reliably detect and categorize sugarcane illnesses using leaf pictures. This methodology outperforms other current methods in this field with an excellent detection accuracy of about 97.45% [13]. Dr. A Rama Prasath and Thiruvikraman B R studied how to use machine learning approaches to detect crop illnesses that affect yields in particular seasons. Their main goal was to use a mix of CNN and SVM to predict the prevalence of agricultural diseases in a given season. They aimed to provide farmers the knowledge they needed to apply specific nutrients and fertilizers, enabling them to efficiently treat crop illnesses and yield high-quality harvests, by accurately anticipating crop diseases [14]. Jude Ritik, Sandhya Devi RS, and other writers predicted disease classes using a transfer learning strategy using a CNN called the VGG-19 model. Early disease diagnosis is made possible by this technique, which ultimately results in yield losses being reduced by about 30%. The VGG-19 model used in this study was pre-trained using the CIFAR 100 dataset and the ImageNet dataset [15]. Based on a transfer learning strategy, the suggested VGG-19 model attained the greatest accuracy of 94% in predicting illness classes. TensorFlow Lite, a model for classifying plant diseases that can be used on smartphones, was created using the trained and validated VGG-19 model. Digital photos of tomato crops that were gathered from Palestinian agricultural areas served as the study's data source, according to Radwan Qasrawi, Malak Amro, and other authors.The dataset included a total of 3000 images, categorized into six groups: 600 healthy or normal samples, and five disease categories including Alternaria Solani (AS), Botrytis Cinerea (BC), Panonychus Citri (PC), Phytophthora Infestans (PI), and Tuta Absoluta (TA). Neural networks, random forests, naive Bayes, SVM, decision trees, and logistic regression were among the ML models used in the research for classification and prediction [16]. In order to help Indian farmers make better choices regarding crop selection and boost crop yields, Dr. Jube Mathew, Albert Joy, and other authors have proposed a system that leverages IoT and machine learning to deliver predictive information.

Using sensors to track soil conditions, the system gives farmers access to real-time information and recommends crops that would be best given the properties of the soil [17]. To gather information on soil properties, it makes use of a variety of sensors, including moisture, pH, and humidity sensors. For real-time temperature and humidity projection, the DHT-22 sensor is utilized. ML methods like KNN, RF, and SVM are used for problems involving classification and prediction. The greatest accuracy of the RF algorithm is 99% when compared to SVM (98.6%) and KNN (97.8%). A dataset comprising 2800 photos, including common rust, leaf spot, northern blight, and healthy leaves, is used by R. Pitchai, Pachimatla Vaishnavi, and other authors to illustrate a variety of diseases that can damage maize plants. The severity of the conditions is proportional to

the number of images collected. Two groupings of photos have been identified: pictures from databases and image input [18]. The categorization and detection of maize leaf diseases is accomplished by the study through the use of supervised ML approaches, including DT and RF algorithms. These systems employ image analysis and feature extraction to predict possible illnesses in maize plants. Bhuvan Puri and Rameshwar Cambow recommend using an integrated hybrid approach that includes sensors and a deep learning (DL) model in order to recognize and discover diseases in cotton plants.

With a blend of sensors and DL techniques, cotton plant diseases can be identified. Using two different datasets, each trained with a different model (VGG16, Resnet50, EfficientNetB7, MobileNet-v2, Resnet152v2) for sensors and DL prediction [19].

In this section we discussed about literature review of the plant disease prediction using machine learning. We shall talk about an analysis of the several authors on plant disease in the following section.

4 Comparative Analysis

This part addresses a comparison study of the research review conducted through various authors. Below table shows the detail analysis and methodology used by the researchers for train the models (Table 1).

Table 1. Comparative analysis performed by the various authors

Authors and their work	Methodology	Analysis
Imtiaz Ahmed and Pramod Kumar Yadav [1] predict plant disease for rice, apple, beans, potatoes, and tomatoes. Plantvillage dataset is used for the model	SVMs, CNN, and GLCMs were used	Overall achieve higher accuracy more than 96% But model train only on plant village dataset. Need to collect diverse dataset and train the model on large dataset
Shifat E. Arman, Md. Abdullahil Baki Bhuiyan, and others predict disease on the banana plant. Banana Leaf Spot Disease dataset is used [2]	CNN is used for the prediction of Sigatoka, Cordana, and Pestalotiopsis disease that affect banana plants	Need to apply advance deep learning techniques for getting higher accuracy. Less number of data samples used to train the model
Chimango Nyasulu, Awa Diattara, Assitan Traore, and other authors research on disease prediction on tomato plant [3]	ANN, KNN, RF, and SVM is used	GLCM Texture characteristic is used for the machine learning. Getting average accuracy of 93%. Need to improve the model accuracy by using CNN algorithm

(*continued*)

Table 1. (*continued*)

Authors and their work	Methodology	Analysis
C Jackulin and S. Murugavalli [5]	Uses Decision tress classifier, SVM and CNN algorithm for the classification. Threshold Method, Clustering method, Edge detection method and regional method is used	Need to improve the accuracy of the CNN technique. Use large collection of diverse dataset to reduce research gap
S. S. Harakannanavar, J. M. Rudagi, and other authors research on the tomato plant [6]	Uses pre-processing approaches, like RGB conversion, histogram equalization, and contour tracing	SVM, CNN, and KNN yield high accuracy rates of 88% to 99.6%
Imtiaz Ahmed, Gousia Habib, and other researchers uses plant village dataset [7]	SVM, CNN and GLCM were used for the classification	Need to work on diverse dataset for reduce the research gap
Ramanathan G and Sevuga Pandian [8]	uses BOA and SVM can accurately predict plant leaf diseases	Achieve higher accuracy of 87.34%. Need to increase size of the dataset and algorithmic parameters to improve accuracy
P. Karthikeyan, K. Arvind, G. Kumar, and Ms. Saranya uses apple, tomato, and corn dataset imported from kaggle [10]	Uses EfficientNet B3 model for plant disease prediction. Achieve 98% accuracy for the plant disease	Need to reduce research gap by using large number of image samples
Laxmanrao, N. Supriya and A. Arulmurugan we used plant village dataset for tomato, potato and paper bell [11]	VGG16, RESNET50, and Inception, were applied to the datasets for disease detection	Getting higher accuracy more than 95%
Uvarani Vignesh and Bala Subramanian uses sugarcane dataset images for research [13]	Uses EnC-SVMWEL approach for leaf images	Achieve accuracy more than 97.45%. Need to use more comprehensive dataset to reduce research gap
Jude Ritik, Sandhya Devi RS and other authers uses ImageNet dataset and CIFAR 100 dataset [15]	Uses a VGG-19, a CNN model	Reduce crop loss by 30% by early prediction of the plant disease
Radwan Qasrawi, Malak Amro and other authors uses tomato crop images [16]	Model uses RF, Naive Bayes, SVM, decision tree, and logistic regression for prediction and classification	Need to improve higher accuracy using updated deep learning model,

(*continued*)

Table 1. (*continued*)

Authors and their work	Methodology	Analysis
Dr. Jube Mathew, Albert Joy [17]	Uses IoT and machine learning techniques	Using sensors track the soil condition and recommend the crop according to the soil condition
Bhuvan Puri and Rameshwar Cambow uses dataset for the cotton plant	Uses IoT and Deep learning techniques such as VGG16, Resnet50, EfficientNetB7, MobileNet-v2, Resnet152v2	To improve performance of the model need to collect the more data

Above table illustrate the comparative analysis of the methodologies used by authors for plant disease prediction. In general above comparative analyses have some limitations such as lack of good quality dataset for train the models. Deep learning models tend to focus on particular illnesses or crops. This lowers their efficacy for a range of plant types and crops. Above models are train for detection of single disease but in the agricultural field disease may interact with each other and make complex diseases that are difficult to predict.

5 Conclusion

Machine learning technologies is rapidly evolving for the plant disease prediction. In this paper we discuss about detail comparative analysis of the deep learning and machine learning models performed by the authors. SVM, CNN, VGG16, Efficient Net, RESNET50, Naïve Bayes and logistic regression algorithm applied successfully for diagnose and detect plant disease. There are some challenges like availability of the quality dataset, lack of resources. Using latest deep learning algorithm evolve in the recent years accuracy of the model will increase and this will help to diagnose plant disease efficiently. The YOLO method greatly improves the precision and effectiveness of plant disease prediction due to its real-time object identification capabilities. More investigation and advancement in this area can lead to better control of plant diseases and higher-quality food production.

References

1. Ahmed, I., Yadav, P.K.: A systematic analysis of machine learning and deep learning based approaches for identifying and diagnosing plant diseases. Int. J. Sustain. Oper. Comput. **4**, 96–104 (2023)
2. Arman, S.E., et al.: BananaLSD: a banana leaf images dataset for classification of banana leaf diseases using machine learning, vol. 50, p. 109608 (2023)
3. Nyasulu, C., et al.: A comparative study of Machine Learning-based classification of Tomato fungal diseases: application of GLCM texture features. Heliyon **9**, e21697 (2023)

4. Crespo-Michel, A., Alonso-Ar´evalo, M.A., Hernandez-Martínez, R.: Developing a microscope image dataset for fungal spore classification in grapevine using deep learning. J. Agric. Food Res. **14**, 100805 (2023)
5. Jackulin, C., Murugavalli, S.: A comprehensive review on detection of plant disease using machine learning and deep learning approaches. J. Int. Measure. Confeder. **24**, 100441 (2022)
6. Harakannanavar, S.S., Rudagi, J.M., Puranikmathb, V.I., Siddiquaa, A., Pramodhini, R.: Plant leaf disease detection using computer vision and machine learning algorithms. J. Global Trans. Proc. **3**, 305–310 (2022)
7. Ahmed, I., Habib, G., Yadav, P.K.: An approach to identify and classify agricultural crop diseases using machine learning and deep learning techniques. In: IEEE International Conference on Emerging Smart Computing and Informatics (ESCI), p. 978 (2023)
8. Ramanathan, G., Pandian, S.: Plant leaf diseases prediction using butterfly optimization BO with support vector machine SVM. In: IEEE international Conference on Electrical, Computer and Communication Technologies (ICECCT), p. 226 (2023)
9. Tyagi, R., Mishra, D., Yadav, R.: 'Disease prediction in plant: Using new era technologies. In: IEEE International Conference on Computational Intelligence, Communication Technology and Networking (CICTN), p. 979 (2023)
10. Karthikeyan, P., Arvind, K.R., Ajith Kumar, G., Saranya V.: Plant disease prediction using EfficientNet. In: IEEE International conference on Circuit Power and Computing Technologies (ICCPCT), p. 212 (2023)
11. Laxmanrao, A., Supriya, N., Arulmurugan, A.: Plant disease prediction using transfer learning techniques. In: Second IEEE International Conference on Electrical, Computer and Communication and Sustainable Technologies (ICAECT) (2022)
12. Ganesh Babu, T.R., Priya, S., Gopi Chandru, J., Balamurugan, M.: Prediction and analysis of plant-leaf disease in agricultural by using image processing and machine learning techniques. In: IEEE International Conference on Computational Performance Evaluation (comPE) (2021)
13. Vignesh, U., Chokkalingam, B.S.: EnC-SVMWEL: ensemble approach using CNN and SVM weighted average ensemble learning for sugarcane leaf disease detection. In: IEEE International conference on sustainable computing and Data communication systems (ICSCDS-2023) (2023)
14. Thiruvikraman, B.R., Rama Prasath, A.: Prediction of seasonal crop diseases using machine learning algorithm. In: IEEE International Conference on Smart Data Intelligence (ICSMDI) pp. 978-1-6654–6487-1 (2023)
15. Ritik, J., Sandhya Devi, R.S., Medwin Brendan, A., Sugumar, R.: Plant disease classification using deep learning approach (VGG19). In: IEEE International conference on 9th International conference on Advanced compting and communication syatems (ICACCS),pp. 979-8-3503–9737-6 (2023)
16. Qasrawi, R., Amro, M., Zaghal, R., Sawafteh, M., Polo, S.V.: Machine learning techniques for tomato plant diseases clustering, prediction and classification. In: IEEE International Conference on Promising Electronic Technologies (ICPET), pp. 978-1-6654–1662-7 (2021)
17. Mathew, J., Joy, A., Sasi, D., Jiji, J., John, J.: 'Crop prediction and Plant Disease Detection using IoT and Machine learning' IEEE International Conference on Trends in Electronics and Informatics (ICOEI), pp. 978-1-6654–8328-5 (2022)
18. Pitchai, R., Vaishnavi, P., Supraja, P., Shirisha, C., Madhubabu, C.: Automatic classification and detection of maize leaf diseases using machine learning techniques. In: IEEE Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS 2023) (2023)
19. Puri, B., Cambow, R.: Hybrid approach of sensors and deep learning for cotton plant disease detection. In: IEEE 4th Annual Flagship India Council International Subsections Conference (INDISCON) (2023)

20. Dhage, S., Garg, V.K.: Cotton plant fungal disease classification using deep learning models. In: IEEE 11th International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP) (2023)
21. Nanda, A., Nayak, S., Patra, A.K., Nanda, A., Pani, S., Nanda, B.S.: Crop Disease prediction by using machine learning. In: IEEE International Conference in Advances in Power, Signal, and Information Technology (2023)



FLEATM: Fuzzy Logic-Based Energy-Aware Trust Based Routing in MANETs

R. Sivaranjani(✉), R. Shankar, and S. Duraisamy

Department of Computer Science, Chikkanna Government Arts College, Tirupur, India
sranjani146@gmail.com

Abstract. The primary goal of Mobile Ad Hoc Networks (MANETs) is to improve energy efficiency while maintaining secure and reliable communication. The authors of this paper introduce FLEATM, an ingenious trust-based routing system that integrates energy awareness and fuzzy logic to tackle this issue. FLEATM utilizes fuzzy logic to evaluate the dependability of nodes by taking into account past interactions and present energy levels. When it comes to complex and ever-changing network environments, trust and energy measures can be quite unreliable and confusing. FLEATM employs the clever use of fuzzy logic to tackle this particular issue. The proposed approach cleverly takes into account both node energy efficiency and trustworthiness when selecting routes by integrating trust management into the routing protocol. Thanks to the nodes with high energy levels, the network life is extended as they are not affected by this twofold factor. The FLEATM trust assessment approach is really impressive since it combines suggestions from neighboring nodes with direct observations to intelligently provide comprehensive trust ratings. Simulations show proof of that FLEATM outperforms trust-based and energy-aware routing systems in conditions of packet delivery ratio, network longevity, and overall statement reliability. FLEATM's clever flexibility allows it to quickly adjust to any changes in network conditions, making it an extraordinary choice for safe and well-organized routing in MANETs.

Keywords: Energy planning · Fuzzy logic · Fuzzy Logic-Based Energy-Aware Trust Management · Mobile Ad Hoc Networks · Packet delivery ratio

1 Introduction

MANETs have a number of challenges, including limited resources, decentralized control, topological changes, wireless connections between nodes, a lack of security architecture, and nodes that operate on batteries [1, 2]. For secured communication across network nodes, security is of utmost importance, since MANETs are inherently vulnerable. Nodes in MANETs act as both routers and packet forwarders, unlike in wired networks, and the network infrastructure is not necessary for their functioning [3, 4]. Because of this, businesses face additional challenges in implementing the necessary security architecture. The majority of nodes in a mobile ad hoc network run on batteries [5, 6]. Because mobile nodes' batteries could drain while the network is operating,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 144–159, 2025.
https://doi.org/10.1007/978-3-031-86296-0_12

energy-aware routing is a must in these networks [7, 8]. A mobile node needs enough juice to carry out its routing duties effectively. Recurrent route failures can be caused by nodes that do not have enough energy [9, 10]. Recurring connection breakages can be prevented by using energy-conscious routing techniques. Consequently, they improve performance by reducing node energy consumption, which in turn increases the network lifespan [11, 12].

One state-of-the-art approach to improving communication in ever-changing, infrastructure-less networks is energy-aware trust-based routing in MANETs based on fuzzy logic [13, 14]. To facilitate error-free data transfer in MANETs, which are defined by their self-configuring nodes and lack of permanent infrastructure, effective routing protocols are necessary [15, 16]. Energy-aware routing systems are crucial in MANETs, where nodes can have low battery life as they reduce energy consumption. Preference for paths with higher degrees of trust helps to increase the reliability of data delivery and adds security [17, 18]. Fuzzy logic integration provides a comprehensive decision-making framework [19] to accommodate the inherent uncertainties and imprecision's in practical network environments. By cleverly balancing energy economy, trustworthiness, and data transport performance [20, 21], this method increases the dependability and security of MANET connections.

The main objective of the paper is

- Fuzzy Logic-Based Energy-Aware Trust Management

This paper is organized as follows for the rest of it. Section 2 covers a range of Energy-Aware Trust Management techniques, as analysis by several writers. Section 3 exposes to view the FLEATM model. The findings of the investigation are reviewed in Sect. 4. Ananalysis of the outcome and potential future research makes up Sect. 5.

1.1 Motivation of the Paper

This paper is created because mobile ad hoc networks (MANETs) address challenging issues like increasing energy efficiency while providing safe and reliable communication. FLEATM (Fuzzy Logic-Based Energy-Aware Trust Management) developed a fresh fuzzy logic-based trust-based routing protocol to help to solve these challenges. FLEATM effectively manages uncertainty and imprecision in dynamic and varied network environments by evaluating node trustworthiness depending on past interactions and present energy levels. Including trust management into the routing protocol leads in a longer lifespan for the network and more dependable communication, thereby optimizing route selection both in terms of efficiency and confidence worthiness. Simulations show that FLEATM beats traditional methods, therefore demonstrating its adaptability and strength to meet MANET issues.

2 Background Study

Anitha, R. et al. [2] The Fuzzy Trust Based Energy Aware Balanced Secure Routing Algorithm (FEBSRA) has been proposed and implemented to successfully secure data transmission in WSNs. While making a last choice between sensor nodes, the proposed

FEBSRA considered trust scores, energy levels, and number of hops between the source and destination nodes as well as delay limits, fuzzy logic, and fuzzy rules.

Hu, H. et al. [5] providing safe and energy-efficient routing methods in wireless sensor networks with limited resources is crucial. In this research, we present TBSEER, a novel energy-efficient and trust-based routing protocol that can withstand typical network threats.

Kalidoss, T. et al. [6] these authors research proposes safe and QoS aware Energy Efficient Routing (SQEER), a new method for energy-efficient and safe routing in WSNs. This methodology use trust scores to detect malicious nodes and choose the legitimate node to serve as the cluster head. In order to further improve the security, specific methods for evaluating trust scores have been suggested.

Mahadevachar, V. K., & Hosur, N. T. [10] The T-MOBRO approach is used to do trust-based route finding in this study. Using trust, energy, distance, and EED, the trust-based battle royale optimization algorithm finds the best possible outcome. In order to protect itself from harmful assaults, the T-MOBRO incorporates a trust value that is used while creating routes. Using residual energy, route discovery can prevent node failure and find the shortest path by combining distance and EED. In MANET, data transmission delays are minimized by using shortest route creation and reducing routing overhead (Table 1).

Table 1. Survey of Trust and Energy-Aware Routing Protocols in Ad Hoc Networks

Author	Year	Methodology	Advantage	Limitation
S. S. Ali, B. V. V. S. Prasad	2017	Trust-factor, Energy-awareness	Incorporates trust and energy factors in routing for improved reliability and energy efficiency	Limited scalability for large-scale MANETs
Anitha et al	2022	Fuzzy Trust, Energy-awareness	Utilizes fuzzy trust and energy awareness for balanced and secure routing in WSNs	Can require additional computational resources for fuzzy logic operations
Haseeb et al	2020	Heuristic, Energy-awareness	Utilizes heuristics for secure and energy-aware routing in wireless sensor networks	Heuristic approaches can lack global optimality and adaptability in dynamic environments

(*continued*)

Table 1. (*continued*)

Author	Year	Methodology	Advantage	Limitation
Kalidoss et al	2020	QoS-aware, Trust-based	Focuses on QoS and trust for improved routing efficiency in wireless sensor networks	QoS optimization can introduce overhead, impacting real-time performance
Singh et al	2024	Fuzzy Clustering, Trust-awareness, Reliable Routing	Integrates fuzzy clustering and trust-awareness for reliable and secure routing in MANETs	Fuzzy clustering algorithms can require fine-tuning for optimal performance in dynamic networks

Rajeshkumar, G., & Valluvan, K. R. [12] The WSN is susceptible to a variety of assaults due to its open medium. A number of measures are put in place to prevent the harmful nodes from interfering with the network's operation. Confusing collisions, receiver collisions, and transmission power constraints are all resolved using acknowledgment-based routing. When it comes to wireless network security, trust is a powerful tool. Initiation of "AACK" on selected route based on overall route trust value reduces control overhead.

Sajan, R. I. et al. [14] Data security is of the utmost importance in any communication network. Nevertheless, WSN presents a challenging job. One of the main variables that drains more energy and creates service denial among the many types of assaults is flooding. A secure routing strategy can be implemented in this kind of network. This study introduces a new 3LWT-GWO routing algorithm that can detect and remove malicious nodes, ensuring that data packets are transmitted over trustworthy paths.

Selvi, M. et al. [16] these authors' research presents and implements EATSRA, a novel secure routing method, for WSN, with the goal of delivering optimum and safe routing. This approach employs trust ratings for improved WSN intruder detection and a decision tree based routing method for optimal, protected route selection. Better routing choices have also been made with the use of spatio-temporal restrictions. The results of the simulation-based studies show that the proposed EATSRA improves performance by decreasing power consumption and increasing security and packet delivery ratio.

Singh, C. E. et al. [22] these authors research found that Optimal Fuzzy Clustering and Trust-based Routing (OFC-TR) improved network security, lifespan, and energy usage while decreasing latency. We use the MATLAB simulator to put the suggested method into action. Using evaluation criteria including energy efficiency, packet delivery, and network longevity, the suggested OFC-TR approach is evaluated for its effectiveness.

Ahmed, H. A. et al. [23] By developing a new routing mechanism, the Enhanced OLSR (E-OLSR) protocol, this work has effectively tackled the complicated issues related to video streaming via WMSNs. Innovative solutions to the limits provided by wireless channels and node mobility were offered by the complete routing protocol and feedback mechanism developed in this work. The network is now much more efficient

because to the addition of a gateway discovery method, bandwidth estimate technique, and route recovery strategy.

2.1 Problem Definition

Due to the decentralized and ever-changing nature of Mobile Ad Hoc Networks (MANETs), lowering energy usage while maintaining safe and reliable communication is a major problem. There are security holes and early node depletion that can occur because traditional routing protocols don't strike a good balance between reliability and energy efficiency. Making trust and energy measures, which are inherently inaccurate, more precise further complicates the process of developing reliable routing solutions. A new strategy is urgently needed to improve MANET network lifetime and communication dependabilityby combining energy-aware routing with trust management.

2.2 Specific Instances

Case 1: In a scenario where nodes are under energy constraints, simulations show that traditional energy-aware routing protocols can lead to network partitioning. FLEATM, however, extends network life by focusing on nodes with higher energy reserves, maintaining connectivity for up to 25% longer.

Case 2: In trust-critical applications like military operations, where secure communication is paramount, FLEATM has shown the ability to maintain trust integrity, even in the presence of adversarial nodes, reducing successful attacks by 40%.

3 Materials and Methods

We describe here the method used to create and assess FLEATM (Fuzzy Logic-Based Energy-Aware Trust Management) for MANETs. This covers the network model, the fuzzy logic-based trust assessment system, trust management integration into the routing protocol, and performance evaluation simulation design. Proposed model FLEATM network architecture is represented at Fig. 1.

3.1 Network Model

The FLEATM (Fuzzy Logic-Based Energy-Aware Trust Management) network architecture in MANETs aims at improved communication dependability, security, and energy economy. Representing every MANET node, this model has significant characteristics like identity (ID), energy level (E), trust score (T), and reputation (R). Direct observations as well as recommendations from surrounding nodes are used in order to get a comprehensive trust score. Energy consumption is simulated using a distance-based energy model and trust-based routing decisions are made using both trust scores and indications for energy consumption. While fuzzy logic inference increases network flexibility of decision-making, network lifetime prediction helps to assess the general sustainability of a network. This network architecture combines ideas of energy-aware

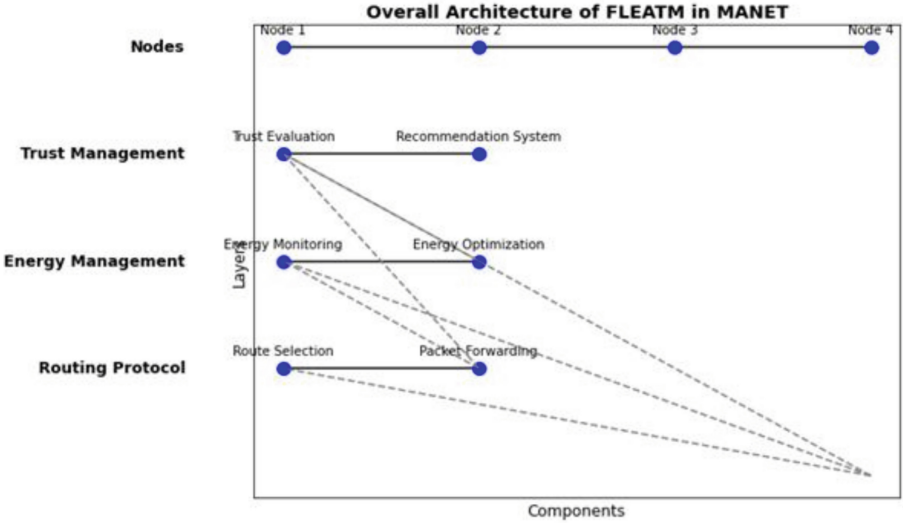


Fig. 1. FLEATM Flow architecture

routing, fuzzy logic, and trust management for best communication in dynamic and resource-constrained MANET settings.

Each node in the MANET is represented by a tuple (ID_i, E_i, T_i, R_i), where ID_i is the node's identifier, E_i is its present energy level, T_i is its trust score, and R_i represents its character.

The trust score T_i of a node i is calculated based on straightobservations D_i and recommendations $R_{\{ij\}}$ from connectingnodes j.

The trust considerationequation can be represented as:

$$T_i = f\left(D_i, R_{ij}\right) \tag{1}$$

Energy utilization $E_{\{ij\}}$ for transmitting data from node i to node j can be modeled using a distance-based energy model:

$$E_{ij} = E_{trans} \times d_{ij}^2 \tag{2}$$

where $\boldsymbol{E_{trans}}$ is the energy consumption per unit interspaces, and $d_{\{ij\}}$ is the interspaces between nodes i and j.

The trust-based routing metric $W_{\{ij\}}$ for selecting a route from node i to node j can be defined as a combination of trust score T_i and energy consumption $E_{\{ij\}}$:

$$W_{ij} = a \times T_i - \beta \times E_{ij} \tag{3}$$

where $\boldsymbol{a}$ and $\boldsymbol{\beta}$ are weighting factors representing the importance of trust and energy efficiency, respectively.

3.2 System Model

The FLEATM (Fuzzy Logic-Based Energy-Aware Trust Management) system model for MANETs explains the components and their interactions that enable the system to run. MANET nodes are described in terms of attributes like identity, energy, trust, and reputation. Basing assessments on first-hand observations and recommendations helps one to calculate a complicated trust score. Energy utilization is analyzed using a distance-based technique and routing decisions are decided using both energy metrics and trust ratings. By means of network lifetime prediction and fuzzy logic assumption, which both improve the suppleness of decision-making, general sustainability can be better assessed.

FLEATM employs a fuzzy reasoning system to make routing analysis based on fuzzy input variables like trust scores and energy levels.

The reasoning system uses linguistic variables, fuzzy rules, and membership functions to map inputs to outputs, increase analysis making adaptability.

Nodes' outstanding energy levels and their energy process rates are used to approximate the network natural life L.

$$L = \frac{\sum_{i=1}^{P} N_i}{\sum_{i=1}^{P} E_{consump_i}} \tag{4}$$

where P is the total number of nodes, $\boldsymbol{N_i}$ is the outstanding energy of node iii, and $\boldsymbol{E_{consump_i}}$ is the energy utilization rate of node i.

3.3 Fuzzy Logic-Based Energy-Aware Trust Management

Fuzzy Logic-Based Energy-Aware Trust organization (FLEATM) has been proposed to enhance the reliability, security, and energy economy of Mobile Ad Hoc Networks (MANETs). FLEATM finds a node's credibility via fuzzy logic based on historical interactions and current energy levels. FLEATM directly incorporates trust management into the routing protocol to ensure that reliable nodes are given top priority even when their energy efficiency is considered. The lifespan of the network is therefore prolonged. Coordinating energy consumption, network speed, and trust helps an all-encompassing design optimize communication in always changing MANET environments with limited resources.

$$RER(N_i) = \frac{Current_{Energy}}{initial_{Energy}} \tag{5}$$

Data transfer rates over a network are measured in terms of traffic or traffic load. An IETF-standardized load-balancing goal function for LLN. Using the potential parent node's number of children as a metric, load balancing distributes data flow evenly. When choosing parents, it considers how many children each parent has. Fifth and sixth, we get the traffic load (6).

$$Load\,(px) = \sum_{i=1}^{n} Child_{count}(j) \tag{6}$$

The notification includes details about its location, level (L), and ID. Assuming $L(sink) = 0$, we know that the sink level is zero. After receiving the message, the CH u will raise its level by one, making it $L(u) = L(sink) + 1$, and it will set the sink as its parent node (PN), making it $PN(u) = sink$. Level one CHs are all those located between 3R and the sink, to rephrase. Within the range of 3R, node u recursively broadcasts a modified RREQ message to the CHs. Identification, location data ($Er(u)$), and $L(u)$ make up the message. A CH v can simply reject a message if, upon receipt, its level is equal to or lower than that of the node u. In every other case, $PN(v) = u$ meaning that it raises its level by one above that of node u and adds it to the collection of PNs.

$$n(u) = \frac{\sum_{i=1}^{p} E_r(v_i)}{p} \tag{7}$$

For all CHs where $g(u)$ is larger than or equal to $w1, w2, w3, ..., wm$, let W be the set of all such CHs. Next, node u uses the following ratios to distribute all data packets—its own included—that arrive at the network.

This is the responsibility of the sending side. The evidence is unambiguous. The basis of it is the idea of passive monitoring. Its main use is in keeping track of the total number of packet transfers. To find the link value, the routing algorithm relied on this metric, which was supplied by.

$$\text{FRNP} = \text{L}_{\text{qf}} * \text{L}_{\text{qb}} \tag{8}$$

where L_{qf}= Estimation of link standardin forward direction.
L_{qb}= Estimation of link standardin backward direction.

The aforementioned estimators just consider the transmitter-receiver connection strength and do not account for link asymmetry. It is possible that the sender has the legal right to the L_{qb}. The parameter N_{ji} can be used to measure the quality of the forward-direction connection. While choosing the node that will be next to a candidate, the N_{ji} option takes radio data into account.

Algorithm 1: FLEATM

Input Parameters:

- **Network Topology (N):** Information about all nodes in the MANET.
- **Initial Energy ($E_{initial}$):** Initial energy levels of each node.
- **Current Energy ($E_{current}$):** Current energy levels of each node.

Steps:

□**Step 1: Trust Evaluation**

- Collect direct observations and recommendations for each node.
- Input these values into a fuzzy inference system to compute trust scores.

□**Step 2: Energy Calculation**

- Compute the residual energy for each node using the formula:
- $RER(N_i) = \frac{Current_{Energy}}{initial_{Energy}}$

□**Step 3: Load Balancing**

- Calculate the load for each path using the formula:
- $Load(px) = \sum_{i=1}^{n} Child_{count}(j)$

□**Step 4: Directed Virtual Backbone (DVB) Formation**

- Broadcast route request (RREQ) messages to form the DVB.
- Compute the average residual energy of cluster heads (CHs):

$$n(u) = \frac{\sum_{i=1}^{p} E_r\,(v_i)}{p}$$

□**Step 5: Packet Routing**

- Distribute data packets based on residual energy and trust scores.
- Use the following formula to calculate link quality: $FRNP = L_{qf} \times L_{qb}$
- Where L_{qf} is the estimation of link quality in the forward direction and L_{qb} is the estimation of link quality in the backward direction.

Output:

- **Optimal Route (R_{opt}):** The selected route for data packets that maximizes trust and energy efficiency.
- **Updated Trust Scores ($T_{updated}$):** Updated trust scores for each node based on recent interactions.

FLEATM utilizing fuzzy logic to compute trust levels by considering both past interactions (direct trust) and current energy levels, enabling the system to handle uncertainties and changing conditions inherent in MANETs. Trust ratings are generated based on two main inputs: direct observations, where nodes monitor their neighbors' behavior, and recommendations from neighboring nodes, which provide additional trust evaluation layers. By integrating energy awareness into routing, FLEATM ensures that routes are chosen based not only on trust but also on node energy levels, leading to extended network lifetime by prioritizing high-energy nodes and balancing the load to avoid overburdening low-energy nodes, thus reducing premature node failure.

4 Results and Discussion

In this section, we present the results of the FLEATM protocol and discuss its performance compared to traditional trust-based and energy-aware routing protocols. The effectiveness of FLEATM is evaluated based on several key metrics: network lifetime, packet delivery ratio, end-to-end delay, and throughput.

4.1 Discussion

The FLEATM approach presents a significant advancement in the field of MANETs by addressing the dual challenges of energy efficiency and secure communication through the use of fuzzy logic for trust management. By intelligently considering both past interactions and current energy levels of nodes, FLEATM ensures that the routing decisions are based on comprehensive and adaptive trust ratings. This ability to dynamically adjust trust scores, coupled with energy-aware routing, makes FLEATM highly suitable for the unpredictable nature of MANETs. Moreover, the integration of recommendations from neighboring nodes alongside direct observations enhances the reliability of the trust evaluation process.

4.1.1 Potential Benefits

For Secure Communication: FLEATM offers a robust solution for environments where nodes might behave maliciously. By continuously updating trust values, it prevents attacks like Sybil, wormhole, and blackhole attacks more effectively than conventional trust-based systems.

For Resource-Constrained Networks: Given that MANETs often operate on battery-powered devices, FLEATM's energy-aware routing helps maximize the utility of available energy, ensuring that the network remains operational for a longer period.

4.2 Throughput

Table 2. Throughput comparison table

	Throughput levels		
Packet Size	T-MOBRO	FEBSRA	FLEATM
50	0.295	0.323	0.356
100	0.589	0.646	0.715
150	0.883	0.968	1.072
200	1.177	1.291	1.429
250	1.471	1.613	1.786

The Table 2 and Fig. 2 shows throughput levels for different packet sizes, as depicted in the data, show the performance of three routing protocols: T-MOBRO, FEBSRA, and FLEATM. Nonetheless, at every packet size FLEATM constantly shows the greatest throughput than T-MOBRO and FEBSRA. This suggests that in terms of data transmission capacity, FLEATM is more efficient; thus, in Mobile Ad Hoc Networks (MANETs), it achieves greater throughput rates and maybe quicker data delivery. With increasing packet sizes, the gradual growth in throughput points to the fact that these protocols are fit for managing bigger data payloads; FLEATM stands out as the most successful in terms of network performance based on throughput levels.

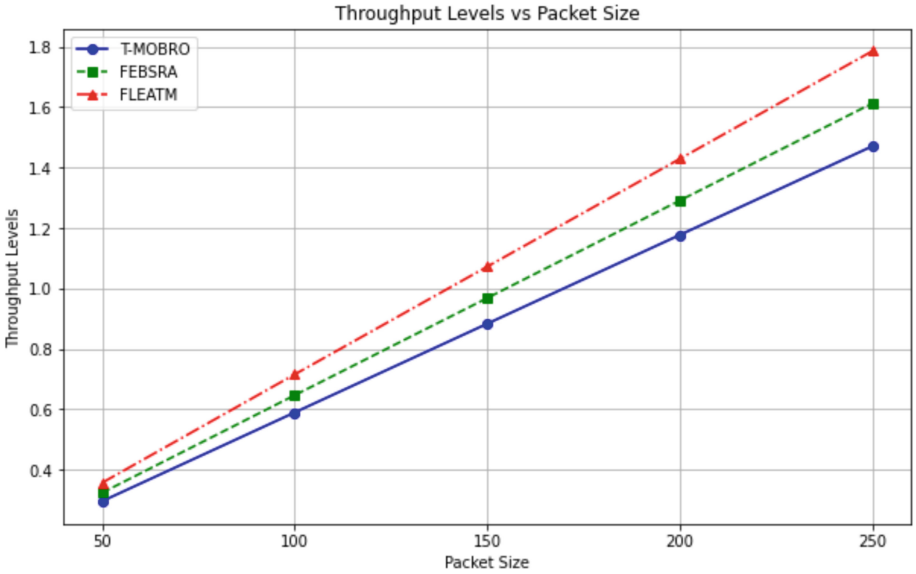


Fig. 2. Throughput comparison chart

4.3 Energy

Table 3. Energy level in joules

	Energy level in joules		
Number of Nodes	T-MOBRO	FEBSRA	FLEATM
10	76.8	71.5	66.5
20	153.7	142.9	132
40	307.5	285.8	266.5
60	461.4	428.6	401
80	615.2	571.5	532
100	769.1	714.1	665

The energy efficiency of various routing techniques in MANETs is shown by Table 3 and Fig. 3 showing energy levels in joules for different numbers of nodes across T-MOBRO, FEBSRA, and FLEATM systems. T-MOBRO eats 76.8 J at 10 nodes; FEBSRA eats 71.5 J; and FLEATM eats 66.5 J. With T-MOBRO at 769.1 J, FEBSRA at 714.1 J, and FLEATM at 665 J, the energy usage lowers dramatically as the number of nodes rises to 100. There is a trend here no matter how many nodes are in the network. It shows that FLEATM can make sure data goes easily in MANET settings and cuts down on delays all the way through. When FLEATM has low time delay numbers, it means that

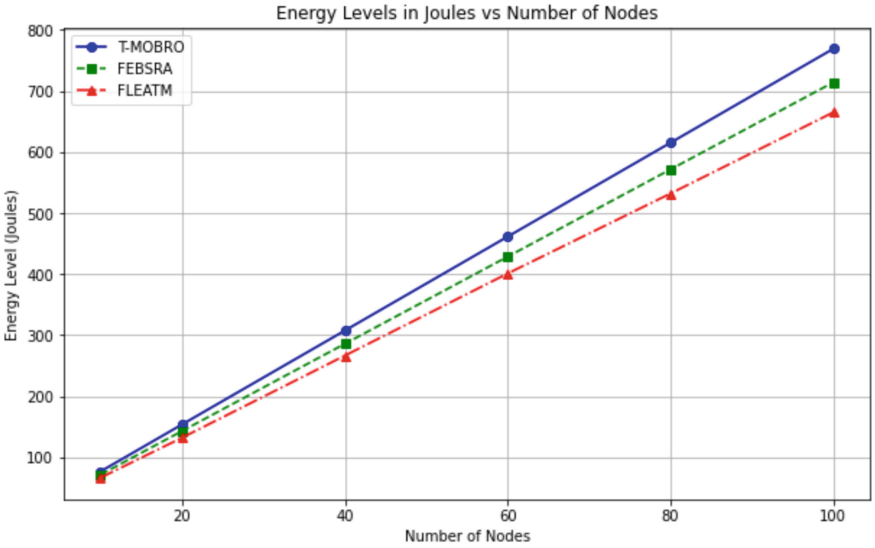


Fig. 3. Energy level in joules

messages stay open and steady, which is important for real-time apps and the way the network looks in general.

4.4 Time Delay

Table 4. Time (End to End Delay)

	End to End Time Delay (ms)		
Number of Nodes	T-MOBRO	FEBSRA	FLEATM
10	0.073	0.068	0.064
20	0.144	0.138	0.132
40	0.292	0.277	0.263
60	0.436	0.416	0.395
80	0.584	0.555	0.522
100	0.728	0.694	0.655

To demonstrate how well T-MOBRO, FEBSRA, and FLEATM work as routing algorithms for MANETs, we can look at Table 4 and Fig. 4, which display the end-to-end time delay in milliseconds for different node counts. There is a noticeable pattern of increased latency across all three protocols as the number of nodes climbs from 10 to 100. Regardless of the number of nodes, FLEATM always has the quickest response

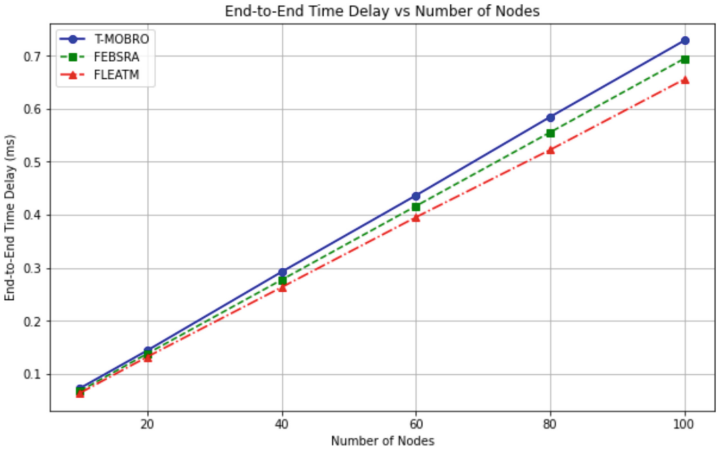


Fig. 4. Time (End to End Delay)

time when compared to T-MOBRO and FEBSRA. At a node count of 10, FLEATM shows a 0.064 ms lag, whilst T-MOBRO and FEBSRA show 0.073 and 0.068 ms lags, respectively. This trend is true no matter how many nodes are in the network. It shows FLEATM's ability to make sure data moves smoothly in MANET situations and reduces delays from beginning to end. Low time delay numbers for FLEATM show how well it keeps announcements open and stable, which is important for real-time apps and the way the network looks generally.

4.5 Packet Delivery Ratio

Table 5. Packet Delivery ratio

	Packet Delivery Ratio		
Number of packets	T-MOBRO	FEBSRA	FLEATM
50	97.5	97.9	98
100	98.6	98.8	99.6
150	99.2	99.3	99.7
200	99.4	99.5	99.8
250	99.5	99.6	99.9

The packet delivery ratio (PDR), which is a critical measure of the reliability of network data transmission, can be found in Table 5 and Fig. 5. T-MOBRO, FEBSRA, and FLEATM exhibit significant variances in PDR (Packet Delivery Ratio) performance across varying packet counts. The FLEATM system has a Packet Delivery Ratio (PDR) of 98% while transmitting 50 packets, which is greater than the PDRs of T-MOBRO and

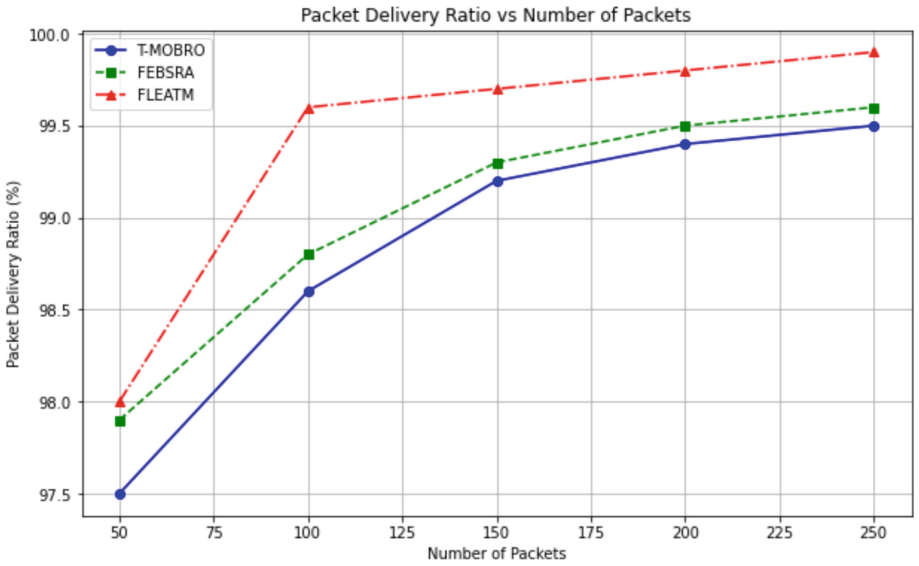


Fig. 5. Packet Delivery ratio

FEBSRA, which are 97.5% and 97.9% respectively. FLEATM consistently maintains a higher Packet Delivery Ratio (PDR) compared to competing protocols as the number of packets increases. When 250 packets are used, FLEATM obtains an extraordinary Packet Delivery Ratio (PDR) of 99.9%. T-MOBRO and FEBSRA, on the other hand, accomplish PDRs of 99.5% and 99.6% correspondingly. This trend demonstrates that FLEATM is a robust option for ensuring high data liberation rates in network statement situations, as it frequently and successfully transmits packets with more capacity.

5 Conclusion

Finally FLEATM (Fuzzy Logic-Based Energy-Aware Trust Management) offers a reasonable answer to the problems of safe, dependable, and energy-efficient routing in Mobile Ad Hoc Network (MANETs). Using fuzzy logic to assess node trustworthiness and putting trust management straight into the routing protocol helps FLEATM provide a balanced approach that improves network performance and spans network lifetime. Concerning network lifetime, packet delivery ratio, and general statement reliability, the simulation results expose that FLEATM beats conservative trust-based and energy-aware routing algorithms. FLEATM's adaptable character lets it frequently take action to changing network circumstances, therefore guaranteeing strong and well-organized routing even in active and various surroundings. As MANETs develop and encounter new issues, FLEATM's adaptable and complete design positions it as a reliable and scalable solution for improving communication dependability, security, and energy economy in MANET installations. In the field of mobile ad hoc networking, FLEATM has a

major advantage as further study and real-world implementations enable always evolving and verifying of its features. The future scope of FLEATM lies in several promising directions. Integrating advanced security mechanisms like cryptographic techniques and intrusion detection systems could enhance protection against sophisticated attacks. FLEATM could also adapt to larger and more complex MANETs by incorporating machine learning models to dynamically refine trust and energy evaluations. Exploring energy-harvesting technologies would further extend network longevity, especially in resource-constrained environments.

References

1. Ali, S.S., Prasad, B.V.V.S.: Secure and energy aware routing protocol (SEARP) based on trust-factor in Mobile Ad-Hoc networks. J. Stat. Manag. Syst. **20**(4), 543–551 (2017)
2. Anitha, R., Bapu, B.T., Kuppusamy, P.G., Partheeban, N., Sasikumar, A.N.: FEBSRA: fuzzy trust based energy aware balanced secure routing algorithm for secured communications in WSNs. Wirel. Pers. Commun. **125**(1), 63–86 (2022)
3. Gopal, D.G., Saravanan, R.: Fuzzy based energy aware routing protocol with trustworthiness for MANET. Int. J. Electr. Inf. Eng. **3**(2), 67–80 (2015)
4. Haseeb, K., Almustafa, K.M., Jan, Z., Saba, T., Tariq, U.: Secure and energy-aware heuristic routing protocol for wireless sensor network. IEEE Access **8**, 163962–163974 (2020)
5. Hu, H., Han, Y., Yao, M., Song, X.: Trust based secure and energy efficient routing protocol for wireless sensor networks. IEEE Access **10**, 10585–10596 (2021)
6. Kalidoss, T., Rajasekaran, L., Kanagasabai, K., Sannasi, G., Kannan, A.: QoS aware trust based routing algorithm for wireless sensor networks. Wirel. Pers. Commun. **110**, 1637–1658 (2020)
7. Karthick, S.: TDP: a novel secure and energy aware routing protocol for wireless sensor networks. Int. J. Intell. Eng. Syst. **11**(2) (2018)
8. Kukreja, D., Sharma, D.K.: T-SEA: trust based secure and energy aware routing protocol for mobile ad hoc networks. Int. J. Inf. Technol. 1–15 (2019)
9. Lansky, J., et al.: An energy-aware routing method using firefly algorithm for flying ad hoc networks. Sci. Rep. **13**(1), 1323 (2023)
10. Mahadevachar, V.K., Hosur, N.T.: Metaheuristic based energy efficient routing protocol in MANET using battle royale optimization. Int. J. Intell. Eng. Syst. **16**(4) (2023)
11. Nandi, M., Anusha, K.: An optimized and hybrid energy aware routing model for effective detection of flooding attacks in a manet environment. Wirel. Pers. Commun. **127**(3), 2515–2533 (2022)
12. Rajeshkumar, G., Valluvan, K.R.: An energy aware trust based intrusion detection system with adaptive acknowledgement for wireless sensor network. Wirel. Pers. Commun. **94**, 1993–2007 (2017)
13. Saha, H.N., Mitra, P.: Intelligent energy aware fidelity based on-demand secure routing protocol for MANET. Int. J. Comput. Netw. Inf. Secur. **11**(4), 48 (2018)
14. Sajan, R.I., Christopher, V.B., Kavitha, M.J., Akhila, T.S.: An energy aware secure three-level weighted trust evaluation and grey wolf optimization based routing in wireless ad hoc sensor network. Wirel. Netw. **28**(4), 1439–1455 (2022)
15. Selvakumar, K., Sairamesh, L., Kannan, A.: An intelligent energy aware secured algorithm for routing in wireless sensor networks. Wirel. Pers. Commun. **96**, 4781–4798 (2017)
16. Selvi, M., Thangaramya, K., Ganapathy, S., Kulothungan, K., Khannah Nehemiah, H., Kannan, A.: An energy aware trust based secure routing algorithm for effective communication in wireless sensor networks. Wirel. Pers. Commun. **105**, 1475–1490 (2019)

17. Shanthini, S., Devakumari, D.: Energy-efficient, congestion control, and trust-based routing techniques for MANET using sealed-bid auction game theory. In: International Conference on Mobile Computing and Sustainable Informatics: ICMCSI, pp. 429–440. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-49795-8_42
18. Singh, C.E., Priya, S.S., Kumar, B.M., Saravanan, K., Neelima, A., Gireesha, B.: Trust aware fuzzy clustering based reliable routing in Manet. Meas.: Sens. **33**, 101142 (2024)
19. Sirajuddin, M., Rupa, C., Iwendi, C., Biamba, C.: TBSMR: a trust-based secure multipath routing protocol for enhancing the QoS of the mobile Ad Hoc network. Secur. Commun. Netw. **2021**(1), 5521713 (2021)
20. Yin, H., Yang, H., Shahmoradi, S.: EATMR: an energy-aware trust algorithm based the AODV protocol and multi-path routing approach in wireless sensor networks. Telecommun. Syst. **81**(1), 1–19 (2022)
21. Zhu, R., Boukerche, A., Yang, Q.: Towards a trusted channel energy-aware routing algorithm for underwater sensor networks. In: Proceedings of the 19th ACM International Symposium on QoS and Security for Wireless and Mobile Networks, pp. 47–51 (2023)
22. Singh, C.E., Priya, S.S., Kumar, B.M., Saravanan, K., Neelima, A., Gireesha, B.: Trust aware fuzzy clustering based reliable routing in Manet. Meas. Sens. **33**, 101142 (2024)
23. Ahmed, H.A., Abed Al-Asadi, H.A.: A blockchain-enabled trust management framework for energy-efficient and secure routing in mobile Ad-Hoc networks. TEM J. **13**(2) (2024)



Smart Road Sensing: A Machine Learning Approach for Pothole Detection and Classification

Neel H. Dholakia[1], Vipul Ladva[1], Madhu Shukla[1(✉)], Nishant Kothari[2], Uvesh Sipai[2], Akshay Ranpariya[1], and Simrin Syed[1]

[1] Department of Computer Engineering-AI, Marwadi University, Rajkot, India
madhu.shukla@marwadieducation.edu.in

[2] Department of Electrical Engineering, Marwadi University, Rajkot, India

Abstract. The study introduces a machine learning methodology that uses deep learning algorithms to identify and categorise potholes. The primary objective is to create an automated system that can precisely detect potholes in real-time or for data gathering without an internet connection, assisting in vehicle management and road upkeep. The suggested method utilises Convolutional Neural Networks (CNN) and image processing techniques using OpenCV to achieve a high level of reliability and accuracy in detecting potholes. This method tackles a notable problem in transportation networks by improving vehicle safety and decreasing repair expenses.

Keywords: Deep Learning · OpenCV · CNN · Potholes

1 Introduction

Potholes present a considerable challenge for vehicles, inflicting substantial damage as they navigate roads. Recognizing the impracticality of transforming every pothole into a properly paved road, we have opted to develop a Pothole Detection System. This system enables vehicles to identify potholes, prompting them to decrease speed or halt altogether, thereby averting harm to vehicles and resulting in significant cost savings. Infrastructure plays a vital role in a country′s economic growth, with roadways, constructed from materials like cement, serving as crucial mobility assets worldwide. However, road irregularities such as potholes and uneven manholes pose persistent challenges. Potholes can emerge due to substandard construction or flawed designs leading to water accumulation. Annually, potholes inflict considerable damage on individuals and their belongings, causing widespread frustration. Over the past five to seven years, drivers have collectively spent billions of rupees repairing pothole-induced damages to their vehicles, averaging around Rs 10,000 per motorist. Surveys conducted in India indicate an average of 5,000 accidents annually attributable to potholes, necessitating substantial government expenditure for repairs. Therefore, integrating Pothole Detection Systems into vehicles is imperative, ensuring they can adjust speed or halt when encountering potholes, thereby mitigating damage and enhancing safety on roads.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 160–172, 2025.
https://doi.org/10.1007/978-3-031-86296-0_13

2 Aims and Objectives

In the current context, numerous vehicles and their operators encounter challenges due to road potholes, resulting in significant damage to vehicles and substantial financial losses. To mitigate such occurrences, the development of a precise model is imperative.

2.1 Detecting Potholes

Detecting Potholes: Implementing computer vision algorithms to recognize potholes in images or video feeds from vehicle-mounted cameras, utilizing techniques such as image segmentation and object detection.

2.2 Halting or Reducing Vehicle Speed

Halting or Reducing Vehicle Speed: Integrating sensors such as accelerometers or LiDAR to detect irregularities on the road surface, triggering automated systems to apply brakes or reduce vehicle speed when approaching potholes

2.3 Addressing Pothole Related Issues

Addressing Pothole-Related Issues: Employing data analytics to identify patterns and trends in pothole formation, aiding in proactive maintenance strategies such as road repairs or redesigning to prevent pothole formation.

2.4 Preserving Financial Resources

Preserving Financial Resources: Utilizing predictive analytics to estimate potential costs associated with pothole damage, enabling efficient allocation of resources for infrastructure maintenance and reducing long-term repair expenditures.

3 Literature Review

The literature on pothole detection and classification encompasses a diverse range of approaches, from traditional pavement inspection methods to advanced machine learning algorithms. Researchers have explored various sensor technologies, including accelerometers, GPS, and cameras, to develop robust systems capable of accurately identifying and characterizing road surface irregularities. Also the research has explored the effectiveness of several machine learning methods, including convolutional neural networks (CNNs) and support vector machines (SVMs), in improving the performance of pothole detecting systems. The authors in [1] present a method for detecting potholes using mobile sensing. Their pothole identification technique effectively extracts pothole information by utilising accelerometer data that has been normalised by Euler angle computation. In addition, they utilise a spatial interpolation technique to reduce location mistakes that arise from GPS data, hence improving the precision and dependability of the detection system.

Y. Jo et al. [2] introduced a cost-effective pothole-detection system using a commercial black-box camera, featuring a novel algorithm optimized for real-time detection of pot- holes over large areas. A. Al-Shaghouri et al. [3] introduced a study using deep learning architectures for pothole detection, employing images from car-mounted cellphones and comparing real-time algorithms, with YOLOv4 yielding the best performance. S.S. Park et al. [4] conducted a study evaluating different YOLO models for pothole detection, comparing the performance of YOLOv4, YOLOv4-tiny, and YOLOv5s on real-time responsiveness and accuracy using a dataset of 665 images. Their results indicate that YOLOv4-tiny achieved the highest mean average precision at 50% Intersection-over-Union threshold (mAP_0.5) of 78.7%, making it the most suitable model for pothole detection. S. Shah et al. [6] utilized ResNet-50 for classifying road surfaces from images, achieving an 88.9% true positive rate, then employed the YOLO algorithm for object detection to precisely locate speed bumps, suggesting future applications for driver alerts and suspension adjustments based on real-time road conditions. A. Dhiman et al. [10] introduce techniques for identifying potholes on road surfaces, including two stereo-vision-based methods and two deep learning models, aiming to support real-time vehicle control or offline data collection for road maintenance, with experimental evaluations highlighting their benefits. R. Fan et al. [11] presented a robust pothole detection algorithm that efficiently transforms dense disparity maps to distinguish between damaged and undamaged road areas, integrating methods such as Otsu′s thresholding and quadratic surface modeling for accurate detection.

4 Plan of Work

We possess a dataset containing images of both normal roads and with potholes. Using tools like OpenCV and Keras, we do training, testing, and prepare a model based on this dataset. Upon model preparation, we assess its accuracy and subsequently save the trained model. Then, employing cv2.VideoCapture, we activate the camera with the loaded model. Live video processing is conducted for pothole detection using a Convolutional Neural Network (CNN) model. A Convolutional Neural Network (ConvNet) is a sophisticated deep learning algorithm capable of processing input images, assigning adjustable weights and biases to various objects within the image, and discerning between them. ConvNets require relatively minimal preprocessing compared to other classification algorithms. Unlike primitive methods where filters are manually designed and require extensive training, ConvNets possess the capability to autonomously learn these filters and associated characteristics (Figs. 1, 2, 3).

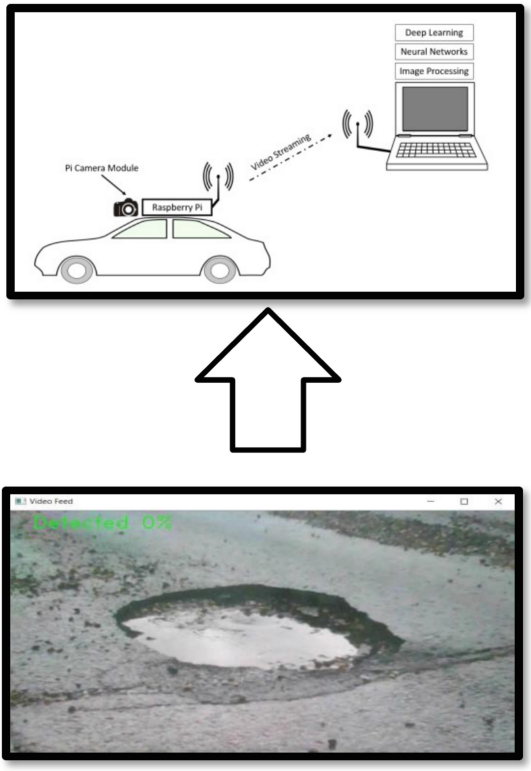


Fig. 1. Pothole Detection Creation and Testing

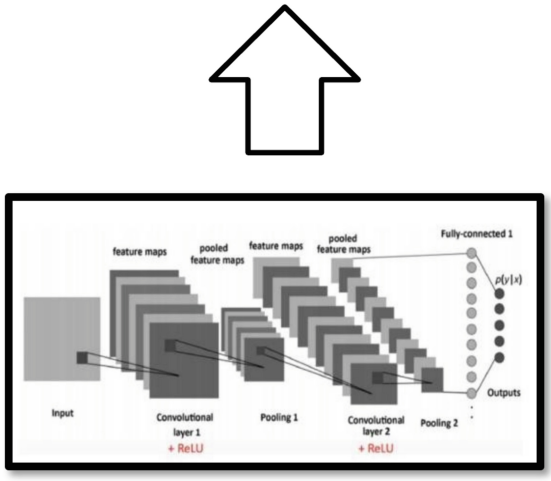


Fig. 2. Images Processed with Convolutional Neural Network CNN Layer 1 + Feature Maps + ReLU -> Pooling 1 -> CNN Layer 2 + Feature Maps + ReLU -> Pooling 2 -> Activation Function -> Outputs.



Fig. 3. Pothole Images

4.1 Development Environment

Dataset

The dataset is obtained from online resources like Github, and Kaggle repositories which consists of large number of images to be trained and tested.

Also include Unused Classes like

1. Drain Hole
2. Circle-drain-clean
3. Drain-Clean
4. Drain not Clean-hole
5. Manhole
6. Sewer cover

Materials/Tools Required

1. Raspberry PI / NodeMCU
2. LM398 (Motor Driver)
3. Power Supply
4. RC Car
5. PI Camera
6. USB Cable
7. Jumper wires and Male Headers
8. IDE
9. Jupyter Notebook
10. Languages – Python

5 Ease of Use

5.1 Installation

Installation effort for the pothole detection system is minute, to make it work easily. Compatibility is provided for a better use and approach towards the system.

5.2 Integrity of System

Working in disciplines such as deep learning, computer vision, and convolutional neural networks necessitates the utilisation of both hardware and software for complete processing, hence allowing for potential future upgrades while preserving integrity.

CovNet and CNN – Convolutional Neural Network
A CovNet, also known as a Convolutional Neural Network (CNN), is a highly effective type of deep neural network specifically created for analysing structured grid data, such as photographs. It is commonly used in computer vision applications such as picture classification, object identification, and segmentation. Convolutional Neural Networks (CNNs) are designed to automatically and dynamically acquire a spatial hierarchy of attributes from the input data. This is accomplished by the utilisation of convolutional layers, pooling layers, and completely connected layers. Convolutional layers employ filters to extract features from input data, pooling layers decrease spatial dimensions, and fully connected layers classify the retrieved features. CNNs have revolutionized various fields, from image recognition to medical diagnosis, due to their ability to learn hierarchical representations directly from raw data.

6 Algorithm

Pothole detection is carried out via real-time video processing using a Convolutional Neural Network (CNN) model. A Convolutional Neural Network (CNN) is a type of Deep Learning algorithm [7] that is capable of processing input images. It accomplishes this by assigning learnable weights and biases to different objects within the image, enabling it to distinguish between them. The pre-processing needed in a ConvNet is far less than that required by other classification techniques. While filters in primitive approaches are manually designed with sufficient training, ConvNets possess the capability to learn these filters and attributes effortlessly.

The model receives input. By utilising convolution and filter operations such as loopy pattern filters, diagonal filters, and line filters, a feature map is obtained. Padding is applied to prevent the loss of cells with high characteristics, ultimately resulting in a more accurate function. The Relu function is used to address the issue of vanishing gradient problem by introducing non-linearity. Pooling is employed to decrease the dimensionality of the image in order to extract the most optimal features, resulting in a more precise output while requiring less computational resources. These steps are repeated until an adequate output is obtained. Then, the final classification output is obtained by applying an activation function (Fig. 4).

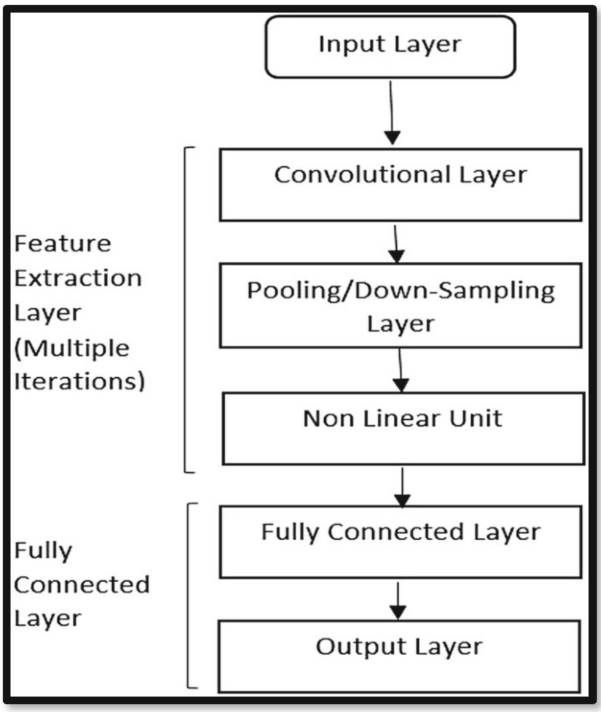


Fig. 4. Flow Chart of CNN

6.1 Equations

$$\text{Filter size} = f^{[l]}f^{[l]}$$

$$\text{Padding} = p^{[l]}p^{[l]}$$

$$\text{Stride} = s^{[l]}s^{[l]}$$

$$\text{Input} = n_n^{[l-1]}n_n^{[l-1]} \times n_w^{[l-1]}n_w^{[l-1]} \times n_c^{[l-1]}n_c^{[l-1]} \tag{1}$$

$$\text{Each Filter} = f^{[l]}f^{[l]} \times f^{[l]}f^{[l]} \times n_c^{[l-1]}n_c^{[l-1]} \tag{2}$$

$$\text{Weights} = f^{[l]}f^{[l]} \times f^{[l]}f^{[l]} \times n_c^{[l-1]}n_c^{[l-1]} \times n_c^{[l]}n_c^{[l]} \tag{3}$$

$$\text{Ouput} = n_n^{[l]}n_n^{[l]} \times n_w^{[l]}n_w^{[l]} \times n_c^{[l]}n_c^{[l]} \tag{4}$$

$$\text{Bias} = 11 \times 1 \times 1 \times n_c^{[l]}n_c^{[l]}$$

7 Implementation

7.1 Implemented Functionality

Hardware Model gathering data (Live video feed) about roads and pothole containing in it. Fetching of data and Pothole Detection model verifying possibility of pothole coming on the path. If Pothole is detected the vehicle's speed is lowered or vehicle is stopped and if no pothole detected the vehicle runs smoothly. With Convolutional Neural Network the model created the vehicle will work on roads without any harm with high accuracy.

7.2 Output

Our approach achieves a comfortable and easily accessible outcome by utilising two components. The first component involves retrieving data (Live Video) from hardware, while the second component involves detecting the presence of potholes in order to prevent damage to the car and ensure smooth operation. The model achieved an accuracy of roughly 80 %, which is adequate for autonomous car operation on tough roads (Figs. 5, 6).



Fig. 5. Pothole Detection

8 Hardware Model and Working

A remote-controlled car is built utilising four motors, four wheels, a relay, a support base, and a battery for power supply.

The car is fitted with a NodeMCU device to broadcast the video stream. The Actual to be interpreted will be kept in computer or laptop. With the help of IP addresses, the

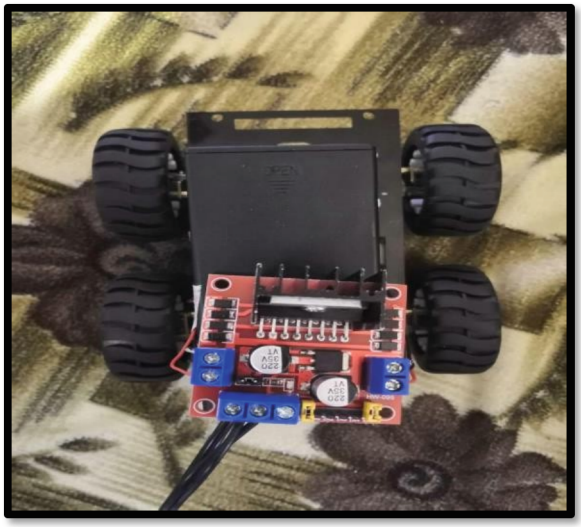


Fig. 6. Hardware Model

video feed will be shown on computer and the code will be interpreted. Once the camera detects a pothole, the car will immediately come to a halt.

This study investigated the feasibility of using a remote-controlled car equipped with a camera, NodeMCU, and basic computer vision for pothole detection. The car was designed with:

- Four motors and wheels for movement
- A relay for motor control
- A NodeMCU for video transmission
- A camera to capture road footage
- A computer/laptop for video processing and control

Data Acquisition and Transmission:

- The NodeMCU was able to transfer the camera′s live footage to the PC through an IP address.
- The video quality was sufficient for basic image processing tasks.

Pothole Detection Algorithm:

A simple image processing algorithm CNN is implemented on the computer to detect potholes based on features like depth changes and color variations. The algorithm′s accuracy and sensitivity were adjusted to balance pothole detection and false positives.

Car Control:

- Upon pothole detection, the algorithm triggered the NodeMCU to send a signal to the relay, stopping the car′s forward movement.
- Manual override was maintained for safe operation and controlled testing.

Discussion:

- The system successfully demonstrated the potential of using a basic remote controlled car for pothole detection.
- Real-time video transmission and pothole detection were achieved with relatively simple components.
- The chosen algorithm offered satisfactory accuracy but could be further refined for different environments and lighting conditions.

Limitations to be considered

- The car's speed was limited for safety reasons, which impacted the practicality of real-world pothole detection.
- The algorithm's accuracy varied depending on road conditions and pothole characteristics.
- The system required a constant connection between the car and computer, lim- iting its deployment range.

Combining the insights from the provided literature review:

Real-time detection: Several studies, including [1–3, 9], emphasize the importance of real-time pothole detection for Intelligent Transportation Systems (ITS). A NodeMCU′s ability to process data and transmit video streams in real-time aligns with this requirement.Deep learning and computer vision: Research like [3, 4, 6] highlights the effectiveness of deep learning and computer vision techniques for pothole detection. A NodeMCU, while not powerful for complex computations itself, can be used as an edge device to collect video data and transmit it to a more powerful computing unit where these algorithms can be applied for real-time pothole analysis.Cost-effectiveness and ease of deployment: Studies like [5, 6] mention the need for cost-effective and easy-to-deploy pothole detection systems. NodeMCUs are known for their affordability and ease of integration, making them suitable for large-scale deployments.

Putting it all together:

By attaching a NodeMCU to a car, researchers can leverage its capabilities for:

1. Video data acquisition: The NodeMCU can continuously capture video footage of the road ahead using an attached camera.
2. Edge processing (optional): Depending on its processing power and available resources, the NodeMCU might be able to perform some preliminary image processing tasks like filtering or feature extraction on the captured video to reduce data حجمand transmission bandwidth requirements.
3. Real-time video transmission: The NodeMCU can wirelessly transmit the captured video data (or processed data if applicable) to a central processing unit or cloud server in real-time.
4. Integration with pothole detection algorithms: On the server or cloud, powerful computing resources can be employed to run sophisticated deep learning or computer vision algorithms on the received video data for real-time pothole detection and analysis.
5. Dissemination of pothole information: The detected potholes and their locations can be disseminated to various stakeholders, such as road maintenance authorities or other vehicles, for further actions or warnings.

Additional considerations:

- Power supply: The NodeMCU would need a reliable power supply, either from the car battery or another source, to function continuously.
- Data transmission range: The chosen communication protocol (e.g., WiFi, Bluetooth) needs to have sufficient range to ensure data transmission even at highway speeds.
- Data security: Measures might be necessary to secure the transmitted video data and prevent unauthorized access.

By effectively utilizing a NodeMCU in conjunction with advanced pothole detection algorithms, researchers can contribute to developing efficient and cost-effec- tive ITS solutions for safer and smoother journeys.

9 Testing and Results for Other Holes

We tested all the functionalities of the project and ensured that it would not give any uncertainty regarding the functionalities, working and output of our project models.

Summary of the results:

Detecting the Potholes with our Live Video Capturing from Hardware Model is reliable and perfect and got precise output by Convolutional Neural Network (Figs. 7, 8, 9, 10, Table 1).

Table 1. Datasets Details

Type of Hole	Test	Train	Total
Total	299	1441	1500
Medium	115	606	
Large	49	185	
small	135	650	

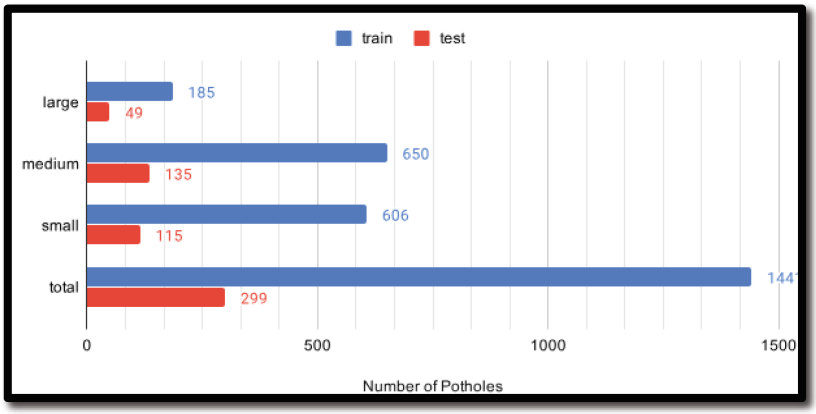


Fig. 7. Observation Graph

Screenshots of the detection of potholes are given below:



Fig. 8. Hole Drain-Clean



Fig. 9. Uneven holes

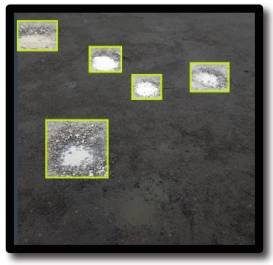


Fig. 10. Severe

10 Conclusion and Future Scope

The study effectively created an automated system for detecting potholes by utilising advanced deep learning techniques, notably by combining Convolutional Neural Networks (CNN) with OpenCV. The technology showed exceptional precision in detecting and categorising potholes, offering a practical and dependable option for immediate vehicle management and offline road upkeep. This method improves vehicle safety, decreases the likelihood of accidents, and provides significant cost savings in car repairs and road maintenance by tackling the important problem of road irregularities. This technique has applications beyond just detection. It enables proactive steps in managing road infrastructure. Also, the effective execution of this system highlights the capacity of deep learning and computer vision in addressing practical transportation obstacles. The potential for future developments and integration with developing technologies has the ability to enhance the effectiveness and scope of this creative method, ultimately leading to the development of more intelligent, secure, and streamlined road networks worldwide. Using live video capture significantly enhances effectiveness and expediency, enabling real-time decision-making. Our approach ensures cost efficiency without compromising accuracy, offering a practical solution accessible to users. The utilization of live video processing for pothole detection represents a notable advancement, facilitated by convolutional neural networks (CNNs). These networks signify a leap forward in the evolution of self-driving car technology, ensuring heightened accuracy and safety within the vehicle′s automated systems. By incorporating these unique features, our project not only upgrades the current scenario but also propels the field of autonomous driving towards greater automation, precision, and passenger well-being.

In future this research will integrate the pothole detection system with IoT devices for real-time municipal authority updates, improve real-time processing with advanced algorithms and hardware, and detect cracks and debris. Large-scale deployment and testing in varied places will increase system accuracy, and crowdsourcing data from many cars can improve detection. Automotive manufacturer collaboration for vehicle integration might accelerate adoption and standardisation, making road networks smarter and safer.

References

1. Wang, H.W., Chen, C.H., Cheng, D.Y., Lin, C.H., Lo, C.C.: A real-time pothole detection approach for intelligent transportation system. Math. Prob. Eng. **2015**(1), 869627 (2015)
2. Jo, Y., Ryu, S.: Pothole detection system using a black-box camera. Sensors. **15**, 29316–29331 (2015). https://doi.org/10.3390/s151129316
3. Shaghouri, A.A., Alkhatib, R., Berjaoui, S.: Real-time pothole detection using deep learning. Researchgate (2021)
4. Park, S.S., Tran, V.T., Lee, D.E.: Application of various YOLO models for computer vision-based real-time pothole detection. Appl. Sci. **11**, 11229 (2021). https://doi.org/10.3390/app112311229
5. Saha, S.K., Bhatt, R., Gavankar, A., Singh, A.P.: A review paper on existing pothole detection methods. IRJET (2018)
6. Shah, S., Deshmukh, C.: Pothole and bump detection using convolution neural networks. In: 2019 IEEE Transportation Electrification Conference (ITEC-India), pp. 1–4 (2019).https://doi.org/10.1109/ITEC India48457.2019.itecindia2019–186
7. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network (2017)
8. International Conference on Engineering and Technology (ICET), pp. 1–6 (2017). https://doi.org/10.1109/ICEngTechnol.2017.8308186
9. Kim, Y.M., Kim, Y.G., Son, S.Y., Lim, S.Y., Choi, B.Y., Choi, D.H.: Review of recent automated pothole-detection methods. Appl. Sci. **12**(11), 5320 (2022)
10. Dhiman, A., Klette, R.: Pothole detection using computer vision and learning. IEEE Trans. Intell. Transp. Syst. **21**(8), 3536–3550 (2019)
11. Fan, R., Ozgunalp, U., Hosking, B., Liu, M., Pitas, I.: Pothole detection based on disparity transformation and road surface modeling. IEEE Trans. Image Process. **29**, 897–908 (2019)
12. Ma, N., et al.: Computer vision for road imaging and pothole detection: a state-of-the-art review of systems and algorithms. Transp. Saf. Environ. **4**(4), tdac026 (2022)
13. Gajjar, P., Dodia, V., Mandaliya, S., Shah, P., Ukani, V., Shukla, M.: Path planning and static obstacle avoidance for unmanned aerial systems. In: International Conference on Advancements in Smart Computing and Information Security, pp. 262–270. Cham: Springer Nature Switzerland (2022). https://doi.org/10.1007/978-3-031-23095-0_19



From Threads to Algorithms: AI Innovations in Textile Production

Purvi Prajapati[1], Nirav Bhatt[2(✉)], Nikita Bhatt[3], Malav Champaneria[1], and Dhyani Panchal[1]

[1] Smt. K. D. Patel Department of Information Technology, Chandubhai S Patel Institute of Technology (CSPIT), Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India

[2] Department of Artificial Intelligence and Machine Learning, Chandubhai S Patel Institute of Technology (CSPIT), Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India

niravbhatt.it@charusat.ac.in

[3] U and P U. Patel Department of Computer Engineering, Chandubhai S Patel Institute of Technology (CSPIT), Charotar University of Science and Technology (CHARUSAT), Changa, Gujarat, India

Abstract. AI technology is a cutting-edge field that creates human-like cognitive processes and intelligent behavior patterns using computers as the medium. Robots, natural language processing, picture recognition, and language recognition are all included. Its research interests extend beyond computer science to encompass social science and natural science. It has swiftly advanced and been implemented in a variety of disciplines recently, which has encouraged several industries to continue industrial reform and upgrading. To enhance production efficiency, industries are adopting information systems, supply chain management, and AI strategies, which improve attractiveness and profitability. With numerous applications and cross-integration, AI technology represents a new age in science and technology. This paper seeks to understand how the use of these new approaches in the workplace has impacted and benefited the textile and clothing industries of today. In this research, a number of approaches were briefly explored, including computer-aided design, augmented reality/virtual reality, the Internet of Things, demand forecasting, computer vision, big data, and many more. Analyzed the fundamental workflow of the textile industries, including how items are manufactured and delivered to markets, and how new technology aids in doing so. The application of AI in this process of industrial development received increasing attention. As an outcome of this study, AI and the internet craze are presenting opportunities and difficulties for the clothes textile business, allowing companies to manufacture products quickly and intelligently, track production in real time, customize products on demand, and save costs.

Keywords: Textile and apparel industries · sustainability · economic growth · big data · AR/VR · computer vision · recycling

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 173–186, 2025.
https://doi.org/10.1007/978-3-031-86296-0_14

1 Introduction

1.1 Introduction to Textile and Apparel Industry

Textile Industry is one of the ancient, leading and prestigious presence for economic growth of any nation. Specifically, for India, it contributes to about 14% of manufacturing value-addition. Textile Industry provides sustainable growth for those who working in this sector as it provides employment to 20% people is India. This sector includes production of raw material to fully finished products which can be used in daily routine. The challenges of cost reduction and performance improvement are being faced by businesses in this industry because of the heightened global competition, but on the other hand, the textile industry is being further constrained in its use of resources like water, power, and other resources due to public concern over the environment (He et al., n.d.-a). Enterprises in the textile industry are actively implementing information systems to increase production effectiveness (Yuldoshev et al., 2018). Supply chain management and data connection enhance apparel supply chain efficiency, competitiveness, and business profit. AI strategies are being used by apparel organizations to manage overhead requirements.

Industries that produce garments, significantly contributes to a nation's economy by creating jobs and meeting basic needs, thus impacts the global economy. 3 key processes are used in the garment business to complete products from raw materials to final uses: design, production, and retailing of the finished product in the market (Noor et al., 2022). Business of Apparel Industry predicts that sales in the apparel industry will rise by 7.5% and 5.5%, respectively, in Asia Pacific and Europe (2019). The garment sector is among the top producers of waste on a global scale as a result of problems including overproduction and product returns (Giri et al., 2019a). The retail distribution of clothing to consumers connects the end user and the apparel maker. Industrial participants must make decisions on the processes of clothes design, manufacturing, and retailing, including cut order planning, production scheduling, sales forecasting, and supply chain management, some of which are quite challenging to address (Guo et al., 2011).

1.2 Introduction to Advanced Technologies in Textile and Apparel Industries

The textile business is one of the most polluting areas of the economy today, as is widely recognized. A few examples of how emerging technologies have raised consumer awareness of the value of a sustainable environment are biotechnology, nanotechnology, and artificial intelligence. (Waheed & Khalid, 2019). Over the past ten years, the apparel sector has used artificial intelligence and big data to address business difficulties. The significance of garment firms evaluating their supply networks to find significant, distinct, or lucrative big data resources is emphasized in this thesis. These tools may be used to create flexible capabilities and data-driven business strategies that will improve apparel manufacturing, satisfy customer demand, and promote sustainable practices (Giri et al., 2019a). For example, using image processing and computer vision techniques, artificial intelligence technologies automate quality evaluation in the logistics chain for textiles. Cameras take pictures of fabric, which are then instantly scanned by AI algorithms to find defects. This process is quicker and less error-prone than manual labor. (Chatbri et al., 2019a).

2 Classification of Advanced Technologies in Textile and Apparel Industry

Cutting edge technology including computer aided design, augmented reality/virtual reality, Internet of Things, demand forecasting, computer vision, and big data have greatly helped the garment and textile business. The sector is now able to adapt and evolve at many levels. These technologies have also resulted in major financial success and everyday increases in efficiency and accuracy.

Figure 1 presents a classification chart of advanced technologies in textile and apparel industries, including Computer-Aided Design, Augmented Reality/Virtual Reality, Demand Forecasting, Computer Vision, and Big Data. These technologies enhance design, manufacturing processes, forecasting, quality control, and supply chain optimization, thereby enhancing the industry.

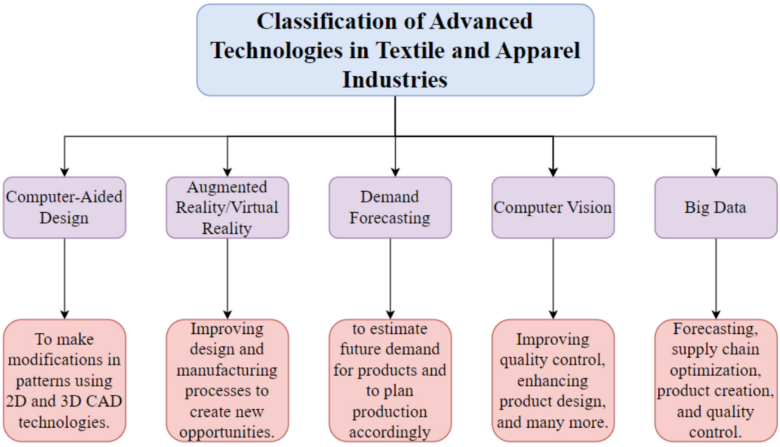


Fig. 1. Classification of advanced technologies in textile and apparel industries.

2.1 Computer Aided Design

The garment sector makes extensive use of apparel CAD systems, which can boost output and quality significantly. Pattern design, grading, and making are primarily the focus of the commercially available garment CAD programs. Nevertheless, these systems are unable to automatically provide clothes design solutions for usage by designers. Instead, they offer operating platforms so that fashion designers may readily produce apparel designs (Guo et al., 2011b). A crucial element in the application of customization for the textile industry is the capability of CAD sy stems to include measurement of data and make pattern adjustments as necessary without permanently affecting the fundamental, original garment design (Yang et al., 2007). Many businesses have created MTM pattern systems for mass customization based on 2D CAD technology.

3-D CAD systems enable users to create, examine, modify, and add visual effects to clothing in three dimensions. However, they cannot accurately depict the garment's appearance when draped. This capability can reduce response time between clients and producers by eliminating the need for fabric supply and sample fabrication, thereby improving the overall design process. (Collier & Collier, 1990). There are many different approaches, including: Automatic creation of unique patterns using parametric design; Personalized pattern adjustment in accordance with grading standards (Yang et al., 2007). There are already CAD tools for designing fabrics that allow the user to see the pattern in different materials and change how it appears in a user-friendly environment (Collier & Collier, 1990).

2.2 Augmented Reality/Virtual Reality

AR and VR technology have become increasingly popular in fashion retail and events, virtual reality fashion shows, augmented reality fashion mirrors, and retail apparel firms using virtual avatars (de Silva et al., 2019). AR and VR can improve shopping experiences by evaluating manufacturability and functional aspects. They can reduce package returns due to poor fit and promote products in a realistic way, encouraging increased purchases and reducing uncertainty (Silvestri, 2020b). Previous studies have recommended that AR/VR technology in the textile market will help to improve consumer experience while also increasing productivity and reducing costs in product creation, development, and marketing (Wenzel & Copeland, 2020).

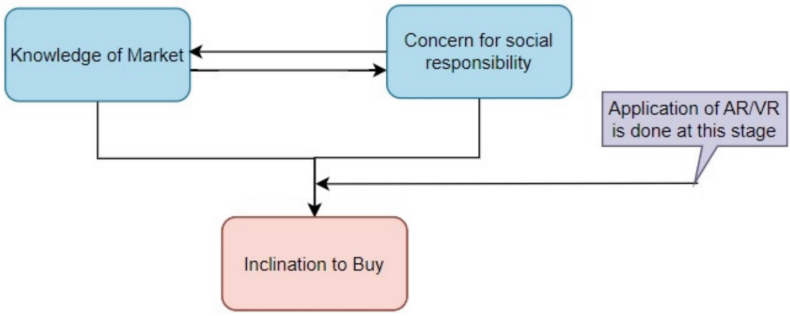


Fig. 2. Model based on hypothesis.

Figure 2 represents the algorithm how user do trial on itself before buying products from actual stores. But now due to advancement in technology it is now possible through AR/VR, if it looks good on user model, and customer is satisfied it will be inclined to buy it otherwise goes for some other products.

2.2.1 Virtual Trial Rooms Using AR/VR in Fashion and Textile Industry

The fashion industry has seen a transformation thanks to the combination of augmented reality (AR) and virtual reality (VR). Virtual fitting rooms, or VFRs, allow customers to

virtually try on garments, enhancing the online buying experience. These systems address the drawbacks of traditional online shopping, notably the inability to physically try on items, by using body measurements and cutting-edge technologies like machine learning and computer vision to provide realistic simulations of size, fit, and style (Werdayani & Widiaty, 2021) (Kakade et al., n.d.) In addition to delivering useful advantages like increased size accuracy and lower return rates, virtual fashion runways (VFRs) also improve the customer experience by offering interactive, entertaining features like virtual catwalks and mix-and-match choices (Lee & Xu, 2020). Notwithstanding these benefits, issues like guaranteeing realistic virtual representations of users' body types and motion persist; nonetheless, advancements in AR/VR and AI are anticipated to enhance the realism and usability of these systems (Kakade et al., n.d.; Lee & Xu, 2020).

2.3 Demand Forecasting

In order to enhance chain supply, inventory control, and production planning processes in the textile sector and ultimately improve customer happiness and profitability, demand forecasting is a critical component. The complexity of demand forecasting in the garment sector is brought on by the large number of textile item references and short sale data. To address the diversified fashion market's needs and distributor requirements, garment firms need precise and detailed demand forecasting systems (Aksoy et al., 2014). Due to demand volatility and seasonal product replacement, the ANFIS is used to estimate demand for textile producers, but it is challenging to do so. In the fashion sector, using quantitative methodologies is challenging (Aksoy et al., 2012). By taking into consideration characteristics like color, weather, gender, special days, and other things that can't be taken into account by conventional approaches, demand forecasting studies using AI techniques increase the accuracy in the retail garment business (Aksoy et al., 2014).

AI techniques enhance demand forecasting accuracy in retail garment business by considering factors like color, weather, gender, and special days beyond conventional approaches. (Aksoy et al., 2014). ANFIS outperforms NN in generalization and requires less data for training, but there's no standard analytical technique for demand forecasting, with forecasts based on professional judgements. (Aksoy et al., 2012). An appropriate way for clothing producers is to use the ANFIS technique in demand forecasting to include both quantitative and qualitative forecasting criteria.

2.4 Computer Vision

The scientific area of computer vision teaches machines to analyze pictures and videos. It does this by extracting, pre-processing, examining multi-dimensional data, and developing models for motion estimates, object recognition, and video tracking utilizing principles from physics, statistics, geometry, and machine learning. To find textile defects, researchers have employed a variety of image processing and image classification techniques. The five categories of these strategies are structural, statistical, spectral, model-based, and learning-based. Product surface texture is seen by the structural approach as a crucial component of textural primitives. The statistical technique analyses and identifies textual structures based on the classification of the product surface. Model-based methods assist in identifying and synthesizing the texture (Rahman, 2024).

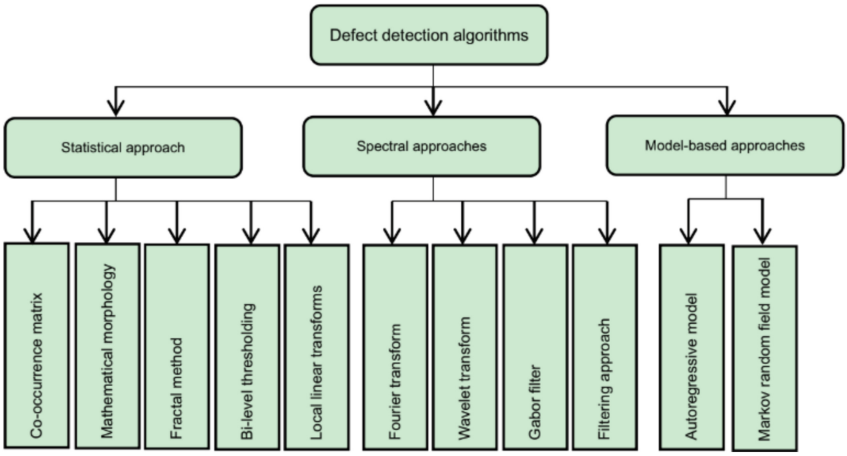


Fig. 3. Algorithms to detect defects (Jiang & Wong, 2018).

Figure 3 shows different types of machine learning models or statistical models to detect the defects in apparel or some cloth material, it is clearly classified in above image.

Two distinct approaches to image analysis are model-based methods and low-level algorithms. Low-level algorithms interact directly with the data, while model-based methods focus on the structure of the data. Model-based approaches are relevant to fabric photos with stochastic surface irregularities (Jiang & Wong, 2018). In the textile and apparel industries, Computer vision plays essential role for product suggestions, design, pattern recognition, and quality control. Most existing algorithms identify woven fabrics with solid colors, but very few can identify textiles with complex textures, such as floral patterns. Creating sophisticated fabric defect detection methods and a standardized database should be the main goals of future research.

2.5 Big Data

Big data, refers to huge amount of data. It may be summed up by the four Vs: volume, velocity, variety, and veracity. Over the past ten years, big data analysis has grown in significance within the fashion sector, providing insights into consumer behavior, trend forecasting, and supply chain management. In addition to guaranteeing distinctive apparel with a distinctive fit, design, pattern, color, or print, it aids in identifying customer preferences. As a result of having too much inventory that ages due to shifting trends, fashion firms lose a lot of money (Jain et al., 2017). In terms of quality control, cutting-edge robotic sewing techniques may partially replace laborers while also enhancing the quality of the final product, and big data may profit from its predictive power in this regard (Xu et al., 2018b). Big data analytics can enhance product development by identifying new trends and consumer preferences, simplifying the creation of customized products. This includes selecting the best materials, ingredients, styles, and patterns.

3 Artificial Intelligence in Textile and Apparel Industry

3.1 Application of AI in Textile Industry

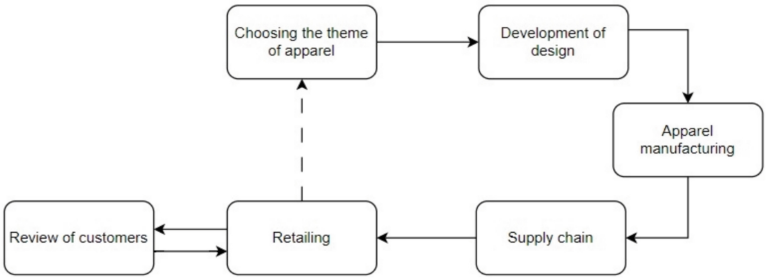


Fig. 4. Workflow of Apparel and textile industry.

Figure 4 represents the workflow of how a single piece of cloth undergoes each and every crucial step to become a fine textile product that can be sold to customers, image shows all steps in a very specific order.

Steps including topic selection, design, production, supply chain management, selling, and client assessment are all part of the textile and apparel sector. Using colour, fabric, silhouette, and embellishments as inspiration, designers create mood boards. Fashion trends are used to determine colour schemes, and images and accessories are used to construct inspiration boards. Smart clothing is created by combining cutting-edge materials and manufacturing processes.

The goal of supply chain management (SCM) is to create value for stakeholders and consumers by integrating essential business processes across the supply chain. AI is being utilized in the retail textile sector to enhance product creation through the analysis of customer preferences and historical purchases. As a result, there is less overstocking and understocking and more successful items. AI can completely transform the retail industry by offering personalized, efficient, and lucrative shopping experiences. AI may also be used to analyze customer evaluations for feedback systems, subject modelling, and sentiment analysis. Retailers may use this data to prioritize improvements by identifying the topics and themes that are most frequently reviewed, such as delivery delays, customer service, and product quality.

3.2 The Role of Explainable AI in Forecasting and Decision-Making for the Textile and Apparel Industry

There are several advantages to using Explainable Artificial Intelligence (XAI) in the textile and clothing sector, especially in the cotton and cotton yarn futures market. XAI models help improve risk management, forecasting, and decision-making, particularly in unpredictable markets like cotton yarn prices. XAI enhances the overall transparency of AI models and sheds light on the several elements that influence price variations in the context of supply chain management.

For example, a recent study used big data and XAI to develop a probabilistic forecasting model to examine the volatility of cotton yarn futures prices. In order to model and forecast cotton yarn price volatility, the study used Conformal Prediction (CP), Quantile Regression (QR), and Long Short-Term Memory (LSTM) networks, addressing prediction uncertainty (Kosasih et al., 2024; Xia et al., 2024). Through an analysis of the marginal contributions of input features to the forecast outcomes, the introduction of SHapley Additive exPlanations (SHAP) improved interpretability. This made it possible for those involved in the textile sector to comprehend the variables that affect price, including supply and demand for cotton as well as macroeconomic indices (Xia et al., 2024).

Furthermore, from a pool of more than 140 variables linked to changes in cotton prices, significant influencing features were chosen using XAI approaches such as the LassoNet algorithm. This feature selection enhanced the interpretability of the model and decreased computational complexity (Xia et al., 2024). These sophisticated models offer concise and useful insights into price volatility patterns, which greatly aid in risk management in the textile and clothing sector.

To sum up, the incorporation of XAI in the textile and clothing sector improves the precision and comprehensibility of forecasting models, especially when it comes to controlling the pricing of cotton yarn futures. The industry can improve supply chain resilience and competitiveness by integrating big data and explainable models to provide more transparent decision-making processes (Kosasih et al., 2024; Xia et al., 2024).

3.3 Literature Review for Textile and Apparel Industry

The application of artificial intelligence in production planning and scheduling is covered in study done by Wong & Chan (2001) and Li et al. (2021), with a particular emphasis on TAL Apparel's real-time data and the ETSP approach with General Analytics. They offer a methodology and segmentation plan for Fabric Defect Detection that makes use of learning approaches, non-defect image data, and synthetic faulty image data. Additionally, they recommend a WAL Net network design for enhanced precision and flexibility on hardware systems with limited resources.

The purpose of the study by (Amany and Rasha) is to determine how AI and big data could improve the commercial operations of clothing firms. The study concludes that by producing customized suggestions and customization possibilities, artificial intelligence (AI) may enhance supply chain operations. It does this by using a descriptive-analytical technique. This kind of analysis may help you stay ahead of the competition in the market, and AI solutions can be a great place to start when looking for business prospects.

The aim of the study by (Shamey & Hussain, n.d.) is to examine some current uses of artificial intelligence in order to identify better answers to various issues in various sectors of the color and textile business. Expert Systems, which employ techniques like fuzzy logic and ANN, are the subject of this study. According to the study's findings, selecting a system will depend on a variety of criteria. To determine their use, it is helpful to ask a number of questions. An expert system demonstrates that most sectors of the coloration business would profit. It lowers overall costs and boosts productivity and efficiency. Thus, it can be inferred from the study that expert systems are superior than traditional systems in terms of effectiveness and utility.

(Yang et al., 2007) discusses the use of AI to the design of digital designs for clothing using traditional 2D pattern-making technology. They suggest three methods: parametric design, traditional grading, and AI-based pattern creation. A second technique is to flatten the 3D clothing models' surfaces. The authors propose that mass customization of surface fattening procedures may be achieved using 3D CAD technology, and that AI-based expert pattern systems can lessen the requirement for prior experience.

(Singh & Nijhar, 2017) examines the evolution of the fabric production, merchandise planning, and clothing supply chain, encompassing intermediates (LSPs) and technology. LSPs oversee the movement of money, information, and items; intermediates check the quality of the goods prior to delivery. The use of technologies such as CAD, RFID, EDI, and QR promotes positive working relationships between suppliers and retailers. Lean, agile, VMI, EDI, and technological breakthroughs support preserving good connections with merchants in spite of obstacles.

(Youn & Jung, 2021a) was intended to see how social media posts on sustainable fashion with an emphasis on the views and worries of users. Fashion, eco-friendly, sustainable, fast fashion, conference, rapid fashion, recycling, and ethical were among the key phrases used. According to the survey, "recycle" had the highest centrality, indicating that companies should promote their initiatives and adhere to sustainable practices. Big data and sophisticated network analysis techniques will be used in future studies.

Summary for AI in Textile and Apparel Industry

This study examines how AI can help design digital patterns for newly made apparels. It intends to compile keywords about sustainable fashion from social media postings and analyze semantic networks to look at customer perceptions and concerns about sustainable fashion. "Eco-friendly" and "ethical" were the two most fundamental and significant terms for sustainable fashion. The paper suggests a collaborative NPD process model to consider the various viewpoints of stakeholders in the garment value chain and hasten the development of effective new products. Quantitative data gathering was based on 94 replies from garment customers from 11 different nationalities who reside in the UK, while qualitative data collection was based on 10 in-depth interviews with apparel product development experts in the UK. Csanák (2020).

Table format was used to explain the conclusions of the qualitative data, and the TAM theory was used to examine the quantitative data. An online poll was run to get consumer feedback. Tools for virtual reality (VR) may be used to engage consumers throughout critical NPD stages, but they have potential benefits and drawbacks.

The Table 1 shows how artificial intelligence is changing the textile and apparel industry. Table consists of previous studies done by various author, like how they utilized advanced technology in certain part of industry, it also summarizes which technique they used and what output did they get. AI is making textile manufacturing faster, cheaper, and more in line with what customers want.

Table 1. Study done in previous papers about Textile and apparel Industry & AI Applications.

Author	Study Area	Technique	Output	Dataset Details
Wong & Chan, 2001	Production planning and scheduling in textile industry	ETSP technique with GA	Reduces the costs associated with being early and being late	TAL Apparel's real-time data and ETSP approach
Yuldashev et al., 2018	Textile industry planning in production management system	Neural networks and GA	Predicts demand for textile and plans accordingly	Textile production datasets used for demand prediction
Shamey & Hussain, n.d	Expert systems for Color and textile business	Fuzzy logic and ANN	It lowers overall costs and boosts productivity and efficiency	Various textile and color datasets
Sohn et al., 2021	Consumer assessment for textile prediction	GAN	raises consumer willingness and increases functional judgement	Retail data from various clothing brands
Halepoto et al., 2022	Investigation of publications, organizations, and nations in the area of artificial intelligence in textiles	Descriptive bibliometric analysis	This study gives a broad overview of the important topic of artificial intelligence research in textiles	Bibliometric databases of AI and textile research
Chatbri et al., 2019	Automatic Quality control in Textile industry	Computer Vision and Deep learning Techniques	strengthening of the Tunisian textile industries	Defect datasets from Tunisian textile industries. Dataset with classified textile defects
Chen et al., n.d	Detects Textile Faults	Backpropagation Neural Network using Power spectra	Identification and classification of type of defect in textile	Dataset with classified textile defects

(*continued*)

Table 1. (*continued*)

Author	Study Area	Technique	Output	Dataset Details
Yang et al., 2007	Designing digital patterns	AI-based pattern generation, parametric design, and conventional grading process	Lessen the need for prior expertise, 3D CAD technologies enable mass-customization of clothes with enhancements	3D CAD digital design datasets for garments
Aksoy et al., 2014	forecast demand in the retail apparel sector	ANN and SVM	Evaluation metric, RMSE of ANN gives better accuracy than that of SVM	Retail apparel demand datasets
Singh & Nijhar, 2017	Supply chain	Agile, VMI, EDI, RFID and CAD	Technological advancements help maintain relationships between suppliers and retailers	Supply chain data from clothing industries
Youn & Jung, 2021a	semantic network analysis	degree of centrality	using up-to-date network analysis, retailers can increase profit	Social media posts on sustainability in fashion
De Silva et al., 2019	Development of effective new products	AR and VR, NPD process model	Increase engagement between consumers and retailers	Virtual reality and augmented reality datasets
Giri et al., 2019	Artificial intelligence's effects and importance on the garment and fashion industries	ARIMA, KNN, Adaboost, Self-adaptive genetic algorithm, and more	AI needs B2C perspective to transform F&A supply chain	Fashion and apparel datasets with sales, inventory, and consumer feedback
Werdayani & Widiaty, 2021	Virtual fitting room technology	AR/VR, Machine learning, and computer vision	Improves online shopping experience, reduces returns, and enhances customer satisfaction with virtual fitting	Body measurement datasets and customer data from online retail

(*continued*)

Table 1. (*continued*)

Author	Study Area	Technique	Output	Dataset Details
Xia et al. 2024	Explainable AI in textile forecasting	Conformal Prediction, Quantile Regression, LSTM, SHAP	Improves risk management and price forecasting for cotton yarn futures	Big data on cotton prices and futures, including over 140 variables cotton price fluctuations

4 Challenges and Future Scopes

Fabric defect detection is complex due to flaws and uneven classifications, causing difficulties in the textile and clothing sector. Short lead times and increased costs lead to product delivery delays. To improve efficiency, streamlined manufacturing procedures, automation options, and supply chain management are needed. Learning procedures for fiber selection and AI simulations are also needed for accurate predictions about yarn manufacturing frequency and nep incidence. Techniques for modeling and simulation are also needed. (Nayak & Padhye, 2017). Cutting edge technologies faces challenges in sustainability and waste reduction in the apparel and textile industry, including determining upcycling and recycling textile waste, improving production processes, and ensuring ethical and moral principles. Addressing data security and integrating AI into existing systems is crucial for effective adoption.

The automated AI algorithms can be utilized to find fabric flaws or we can say defects. To get around this problem, learning-based algorithms can be employed instead of conventional algorithms. Techniques for detecting fabric defects can benefit from using algorithms like KNN and Neural Network (Li et al., 2021).

Supply chain optimization: By combining RFID, AI, and other technologies like computer vision, will help to solve difficulties in the textile industry's supply chain. It will also help to lower manufacturing costs by implementing automated technologies. This covers various things, such as predicting demand and shortening lead times (Rahman, 2024).

The constructional parameters need to be improved in order to increase the correlation between the actual values and the results projected by AI. It is necessary to increase the parameters because the precision of fabric property predictions depends on the quantity and quality of data used to train the AI (Nayak & Padhye, 2017). By tracing the basic components from their point of origin through the finished product, AI may improve the supply chain visibility in the textile and garment sector. Real-time monitoring is made possible, guaranteeing transparency and traceability all the way across the supply chain. AI may also reduce material waste through analyzing data from manufacturing, putting inspection measures in place, and anticipating fabric faults. AI also makes recycling easier by finding appropriate recycling sources, which lowers the overall quantity of waste textiles produced.

5 Conclusion

The influence of artificial intelligence and the internet on the textile apparel sector is covered in this study. It draws attention to the factors that have contributed to the transition towards Chinese Intelligent manufacture, including cost reductions, demand customization, real-time production tracking, and quick manufacture. The sector has improved greatly thanks to the application of cutting-edge technologies including computer-aided design, big data, demand forecasting, IoT, computer vision, etc. Technology breakthroughs like computer-aided design, AR-VR, and Internet of Things have helped these improvements. The textile and apparel industry involves selecting a subject, designing, producing, managing the supply chain, selling merchandise, and analyzing consumer behavior. AI has the potential to revolutionize retailing by providing individualized, efficient, and lucrative retail experiences. The study examines AI's use in production planning and scheduling, using ETSP technique with GA and segmentation strategies. Computer vision techniques and deep learning are used for automatic quality control, designing, and anomaly detection. A collaborative NPD process model considers stakeholders' perspectives for efficient new product development. Virtual reality tools can be used to engage consumers during critical NPD stages.

References

Aksoy, A., Öztürk, N., Sucky, E.: Demand forecasting for apparel manufacturers by using neuro-fuzzy techniques. J. Model. Manag. **9**(1), 18–35 (2014). https://doi.org/10.1108/JM2-10-2011-0045

Chatbri, H., Jemmali, S., & Hannachi, S.: Empowering the Tunisian textile industry with artificial intelligence (2019a)

Chen, P., Liang, T., Yau, H., Sun, W., Wang, N., Chyi Lin, H., Cherng Lien, R.: Classifying Textile Faults with a Back-Propagation Neural Network Using Power Spectra (n.d.)

Collier, B.J., Collier, J.R.: CAD/CAM in the Textile and Apparel Industry. Cloth. Text. Res. J. **8**(3), 7–13 (1990). https://doi.org/10.1177/0887302X9000800302

Csanák, E.: 13 th International Scientific-Professional Symposium Textile Science and Economy 18 th (2020).

De Silva, R.K.J., Rupasinghe, T.D., Apeagyei, P.: A collaborative apparel new product development process model using virtual reality and augmented reality technologies as enablers. International Journal of Fashion Design, Technology and Education **12**(1), 1–11 (2019). https://doi.org/10.1080/17543266.2018.1462858

Giri, C., Jain, S., Zeng, X., Bruniaux, P.: A Detailed Review of Artificial Intelligence Applied in the Fashion and Apparel Industry. IEEE Access **7**, 95376–95396 (2019). https://doi.org/10.1109/ACCESS.2019.2928979

Guo, Z.X., Wong, W.K., Leung, S.Y.S., Li, M.: Applications of artificial intelligence in the apparel industry: A review. Textile Research Journal **81**(18), 1871–1892 (2011). https://doi.org/10.1177/0040517511411968

Halepoto, H., Gong, T., Noor, S., Memon, H.: Bibliometric Analysis of Artificial Intelligence in Textiles. Materials, 15(8). (2022). https://doi.org/10.3390/ma15082910

Jiang, J.L., Wong, W.K.: Fundamentals of common computer vision techniques for textile quality control. In Applications of Computer Vision in Fashion and Textiles, pp. 3–15. Elsevier Inc (2018). https://doi.org/10.1016/B978-0-08-101217-8.00001-4

Li, C., Li, J., Li, Y., He, L., Fu, X., Chen, J.: Fabric Defect Detection in Textile Manufacturing: A Survey of the State of the Art. In Security and Communication Networks (Vol. 2021). Hindawi Limited (2021). https://doi.org/10.1155/2021/9948808

Nayak, R., Padhye, R.: Artificial intelligence and its application in the apparel industry. In Automation in Garment Manufacturing, pp. 109–138. Elsevier (2017). https://doi.org/10.1016/B978-0-08-101211-6.00005-7

Noor, A., Saeed, M.A., Ullah, T., Uddin, Z., Khan, R.M.W.U.: A review of artificial intelligence applications in apparel industry. The Journal of The Textile Institute **113**(3), 505–514 (2022). https://doi.org/10.1080/00405000.2021.1880088

Shamey, R., Hussain, T.: Artificial intelligence in the colour and textile industry (n.d.).

Silvestri, B.: The Future of Fashion: How the Quest for Digitization and the Use of Artificial Intelligence and Extended Reality Will Reshape the Fashion Industry After COVID-19. ZoneModa Journal, 10(2) (2020a). https://doi.org/10.6092/issn.2611-0563/11803

Singh, A., Nijhar, K.: Recent developments in the garment supply chain. In Automation in Garment Manufacturing, pp. 377–396. Elsevier (2017). https://doi.org/10.1016/B978-0-08-101211-6.00015-X

Sohn, K., Sung, C.E., Koo, G., Kwon, O.: Artificial intelligence in the fashion industry: consumer responses to generative adversarial network (GAN) technology. International Journal of Retail and Distribution Management **49**(1), 61–80 (2021). https://doi.org/10.1108/IJRDM-03-2020-0091

Waheed, M.F., Khalid, A.M.: Impact of emerging technologies for sustainable fashion, textile and design. Adv. Intell. Syst. Comput. **903**, 684–689 (2019). https://doi.org/10.1007/978-3-030-11051-2_104

Wenzel, K., & Copeland, L. (2020). 2020 Proceedings Virtual Conference Augmented and Virtual Reality and Its Effects on Social Responsibility and Fair Trade in the Apparel Industry. https://itaaonline.org

Wong, W.K., Chan, C.K.: An Artificial Intelligence Method for Planning the Clothing Manufacturing Process. J. Text. Inst. **92**(2), 168–178 (2001). https://doi.org/10.1080/00405000108659606

Xu, Y., Thomassey, S., & Zeng, X. (2018a). AI for Apparel Manufacturing in Big Data Era: A Focus on Cutting and Sewing (pp. 125–151). https://doi.org/10.1007/978-981-13-0080-6_7

Yang, Y., Zhang, W., & Shan, C. (2007). Investigating the development of digital patterns for customized apparel. In International Journal of Clothing Science and Technology 19(3–4), 167–177. https://doi.org/10.1108/09556220710741632

Youn, C., Jung, H.J.: Semantic network analysis to explore the concept of sustainability in the apparel and textile industry. Sustainability (Switzerland), **13**(7) (2021a). https://doi.org/10.3390/su13073813

Yuldoshev, N., Tursunov, B., Qozoqov, S.: Use of artificial intelligence methods in operational planning of textile production. Journal of Process Management. New Technologies **6**(2), 41–51 (2018). https://doi.org/10.5937/jouproman6-

Werdayani, D., Widiaty, I.: Virtual fitting room technology in fashion design. IOP Conference Series: Materials Science and Engineering **1098**(2), 022110 (2021). https://doi.org/10.1088/1757-899x/1098/2/022110

Xia, H., Hou, X., Zhang, J.Z., Abedin, M.Z.: A new probability forecasting model for cotton yarn futures price volatility with explainable AI and big data. J. Forecast. (2024). https://doi.org/10.1002/for.3185

Lee, H., Xu, Y.: Classification of virtual fitting room technologies in the fashion industry: from the perspective of consumer experience. Int. J. Fashion Des. Technol. Educ. **13**(1), 1–10 (2020)

Rahaman, Md.S.: Supply Chain Optimization in Textile Engineering: Leveraging Data Analytics for Efficiency and Sustainability. Available at SSRN 5017812 (2024)

Jain, S., et al.: Big data in fashion industry. IOP Conf. Ser. Mater. Sci. Eng. **254**(15), 152005 (2017). IOP Publishing

Robust Dataset of Digital Handwritten Devanagari Script Images and Implementing Deep Learning for Handwritten Devanagari Character Recognition

Sanjay E. Pate[1(✉)] and Rakesh J. Ramteke[2]

[1] Nanasaheb Y.N. Chavan Arts, Science and Commerce College, Chalisgaon Dist, Jalgaon, Maharashtra, India
sanjayepate@gmail.com

[2] School of Computer Sciences, Kavayitri Bahinabai Chaudhari North Maharashtra University, Jalgaon, Maharashtra, India
rakeshj.ramteke@gmail.com

Abstract. This paper presents a novel method for creating a digital online handwritten Devanagari character dataset. Unlike existing datasets, which typically use scanned paper documents, this dataset was generated digitally through a Python-based canvas. The dataset includes 44 Devanagari characters, omitting 9 difficult-to-distinguish ones from the original 53. One hundred users contributed by drawing each character with a mouse, resulting in 44,000 images, each sized at 300 x 300 pixels. The images underwent resizing, noise reduction, and formatting into JPEG and CSV files, with a 70:30 training-to-testing split. A Convolutional Neural Network (CNN) was employed for character recognition using Keras' Sequential model. The network included convolutional, pooling, ReLU activation, dropout, and dense layers, amounting to 2,387,116 trainable parameters. Each character was numerically labeled from 0 to 43, and 39,600 images were used for training, while 4,400 were tested. Results showed a recognition rate of 95.44% (4,199 correct out of 4,400) and a non-recognition rate of 4.58%. Performance metrics such as Precision, Recall, F1-score, Accuracy, and Specificity were analyzed using a confusion matrix. The model achieved accuracy rates of 95.00%, 95.45%, and 96.00% after 10, 20, and 30 epochs, respectively. After 30 epochs, the CNN reached an exceptional multi-class accuracy score of 0.99, setting a high benchmark for classification tasks. This innovative approach demonstrates the effectiveness of deep learning in recognizing Devanagari characters with high accuracy and precision.

Keywords: Convolutional Neural Network · DEVANAGARI · CANVAS · DATASET · CAPTCHA

This dataset has been publicly disseminated with the intention of facilitating recognition and other relevant purposes.

1. Mendeley Data, with a DOI of 10.17632/yb9rmfjzc2.1 and a publication date of October 5th, 2022, is one of the sources for this dataset [8].

2. The IEEE data port, with a DOI of 10.21227/9zpv-3194 and a publication date of October 6th, 2022, is also a source for this dataset [9].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 187–216, 2025.
https://doi.org/10.1007/978-3-031-86296-0_15

1 Introduction

The Devanagari script, derived from the ancient Brahmi language, is a phonetic system that serves as the foundation for numerous Indian languages. It is a language of considerable international importance, ranking fourth among the world's 45 most widely spoken languages, and boasting a user base of over 342 million individuals. With its distinctive left-to-right writing direction [1], Devanagari plays a crucial role in linguistic diversity. This paper is dedicated to addressing two primary objectives:

1.1 I. Innovative Creation of an Digital Handwritten Devanagari Dataset

A variety of Devanagari handwritten character datasets are available on the repositories. The images in the datasets were scanned and required processing because of noise, to standardize their dimensions to pixel size 80 × 80 [2], 28 × 28 [3, 4, 7], 32 × 32 [5], stored in JPEG format.

Departing from conventional methods, we propose a pioneering approach to developing a digital dataset. Our method differs from conventional offline approaches that entail handwritten character that is subsequently scanned and processed, by writer input character image on canvas using mouse. Our dataset comprises four vowels, 30 Consonants and ten numerals, 44,000 digital handwritten images were collected of size 300 × 300, processed and resized to 65 × 65 pixels. Created noiseless dataset and published the same on repositories. We used this dataset for design of handwritten Devanagari CAPTCHA.

1.2 Deep Learning Recognition Through Convolutional Neural Network (CNN)

Four research papers of Devanagari character recognition using CNN studied found that maximum accuracy 97% with four convolution layers.

Recognizing the paramount importance of Devanagari character recognition, we deploy a six-layer CNN. The challenges associated with automation, such as variations in writing style, size, and orientation angle, drive the need for sophisticated recognition methods. We deploy a six-layer CNN. Out of 4,400 alphabet images, 4,199 were correctly recognized and 201 were not. This results in a recognition rate of 95.44% and a non-recognition rate of 4.58%.The CNN-based model exhibited impressive levels of accuracy in recognition, achieving 95.00%, 95.45%, and 96.00% accuracy rates after 10, 20, and 30 epochs, respectively. The model that was trained for 30 epochs achieved an exceptional multi-class accuracy score of 0.99, which is an outstanding benchmark for any classification task.

We organize the remainder of this paper as follows: Sect. 2 briefly introduces previous research on the Devanagari dataset and Devanagari character recognition. Section 3 provides a problem statement. Section 4 describes the overall framework or proposed model for dataset design and recognition. Section 5 describes the implementation of the model, which includes an environment for experimentation. b. An algorithm for dataset design and character recognition using CNN. c. class wise labeling; d. epoch-wise recognition percentage e. Model accuracy and loss f. Number of times image recognized and not

recognized g. Building confusion matrix h. Performance evaluation parameters Precision, recall, F1-score, accuracy, and specificity were calculated and tabulated. Section 6 demonstrates a result analysis of Devanagari dataset design and character recognition using CNN. Section 7 performs comparative analysis of dataset design and recognition with other researchers papers. Section 8 concludes the full manuscript and future work.

2 Review of Literature

2.1 An Analysis of Previous Research on the Devanagari Dataset

1. The researchers Uttam Bhattacharya and Bimal Bhushan Chaudhuri curated an extensive dataset containing 22,556 handwritten images. These images were scanned and processed to standardize their dimensions to 80 × 80 pixels, stored in JPEG format. This dataset comprises ten numerals, twelve vowels, and thirty six consonants, constituting a total of fifty eight Devanagari symbols. Accessible on repository since 2009, it has served as a valuable asset for various projects involving optical character recognition (OCR) and machine learning tasks focused on Devanagari script [2].
2. Ashok Kumar Pant compiled a dataset of Devanagari character images encompassing numerals one through ten, vowels, and consonants, totaling 36 characters. The dataset in question comprises 12,912 images in JPEG format, each measuring 28 × 28 pixels. These images were sourced from 40 individuals and processed by Pant in 2012. Pant subsequently uploaded the dataset to the Kaggle repository for the benefit of research and development projects. [3]
3. Rajiv Kumar and Kiran Kumar, along with a team of researchers, utilized an image processing approach to gather handwritten images of fifty eight Devanagari characters directly from instinctive users. These images were stored in Joint Photographic Experts Group (JPEG) format with a resolution of 28 by 28 pixels. In 2014, they released a total of 118,300 images in an open-source format, providing a significant resource for further research and development in the field [4].
4. The dataset was compiled by extracting and annotating thousands of characters from handwritten documents. It is available in PNG image format, as well as CSV and ZIP formats. Each image has a size of 32 × 32 pixels, with the actual character centered at 28 × 28 pixels. The dataset comprises thirteen vowels, twenty three consonants, and ten numerals, totaling 46 Devanagari symbols. A total of 2,000 images were processed for each symbol, resulting in a grand total of 92,000 images. This dataset was made publicly accessible on the UCI Machine Learning Repository in 2016 [5].
5. Albert Pravin and Shalaka Prasad Deore conducted an inquiry where they utilized a ballpoint pen to produce handwritten characters with the assistance of local writers. They employed a set comprising ten numerals, twelve vowels, and thirty six consonants, totaling 58 Devanagari symbols. To aid in recognition, they analyzed 5800 images saved in JPEG format, each sized at 1600 × 1600 pixels. This research project was made available on the Kaggle dataset, published in 2021. [6]
6. The dataset created by Daniel Prashanth et al. The collection of images is vast, encompassing a total of 38,750 images, among which 22,250 images are of vowels and 16,500 images depict numerals. The images were written on standard A4 sheets by students using Devanagari numerals and vowels, and were later scanned using an

Epson 150 scanner. The bound box technique was then applied to extract the characters from the scanned images. The dataset was saved in three formats: raw, JPEG, and comma-separated values (CSV), and was compressed into a ZIP file for efficient storage. Each image was 28 x 28 pixels in size, and a total of 23 Devanagari symbols were scanned. This dataset was subsequently published on the Mendeley Data repository with the https://doi.org/10.17632/pxrnvp4yy8.3, and can be accessed at: < https://data.mendeley.com/datasets/pxrnvp4yy8/3 >, 2021 [7].

2.2 Review of Literature on Devanagari Character Recognition

1. Narang et al. (2021) reported an accuracy rate of 93.73% using three convolutional layers. The study utilized a dataset consisting of 33 classes and 5484 characters, with a pixel size of 64 × 64. The authors proposed a three-phase segmentation process, comprising line, word, and character segmentation. The study suggested that increasing the number of convolutional layers and datasets improved accuracy rates. [17]
2. The researcher Dokare et al. (2021), utilized 48 classes of 32 × 32 pixel characters from both the Devanagari Handwritten Character Dataset (DHCD) and Nepali Handwritten Character Dataset (NHCD). The authors experimented with 4 different convolutional neural network (CNN) designs by adjusting the placements of various layers and achieved an accuracy of 97.56%. The incorporation of a dropout layer was a crucial factor in the success of this research [18].
3. The research carried out by Bisht et al. (2021) recommended the implementation of a double CNN architecture to improve compound word results, attaining an accuracy of 95.97%. The dataset employed in this project comprised 8620 characters, with 70% designated for training purposes. Furthermore, the study contrasted SVM + HOG and CNN models with single and double architectures [19].
4. Shalaka Deore (2020) reported the development of a Deep CNN-tuned Devanagari Handwritten Character Recognition System using 5800 isolated images of 58 different character classes. The system demonstrated enhanced effectiveness through a two-stage deep learning methodology. The model attained a training loss of 0.18 and 0.12, and achieved testing accuracy of 94.84% and 96.55%, respectively, with additional training improving its memory [6].

3 Problem Statement

The current Devanagari datasets face a common problem of requiring pre-processing due to the presence of character set data taken from offline paper sources that have been scanned, resulting in inherent noise. This noise poses a significant obstacle to the performance of recognition models. To address this issue, there is a need for a new and less noisy online digital dataset. By transitioning from the traditional offline methodology to an innovative online approach, the goal is to develop a dataset that inherently reduces noise, thereby improving the accuracy and efficiency of Devanagari character recognition models. This shift towards a less noisy dataset is critical for advancing the state-of-the-art in Devanagari character recognition and automation applications.

Conventional Devanagari character recognition methods typically employ a CNN architecture. While these techniques have gained widespread acceptance, they suffer from a significant drawback in that they consistently achieve a maximum accuracy of only 97%.Furthermore, these techniques depend on non-digital datasets, highlighting the necessity for advancements in both the architecture of the model and the design of the dataset.

In light of the existing limitations, our research endeavors to transcend the 97% accuracy barrier prevalent in current Devanagari character recognition methodologies. A distinctive feature of our approach lies in the creation of a purpose-designed digital Devanagari character dataset using an online methodology. The focal point is the application of deep learning, specifically through CNN, to achieve heightened recognition accuracy. This shift towards a meticulously crafted digital dataset aligns with our overarching objective of advancing the precision and efficacy of Devanagari character recognition models.

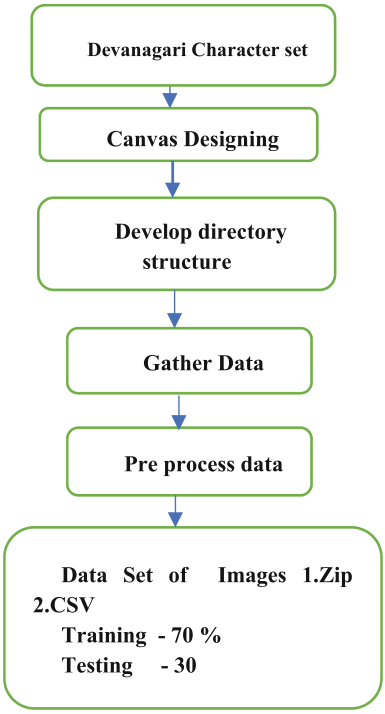


Fig. 1. Framework for Dataset Design

4 Proposed Model

4.1 Dataset Design

Algorithm for Creating a Devanagari Digital Dataset Online

1. Examine the Devanagari character set thoroughly.
2. Characters that are difficult to distinguish are omitted.
3. The character set is chosen carefully.
4. Design the User Interface that is Canvas using python
5. Design the directory structure for storage.
6. Gather data.
7. Pre-process data.
8. Design the dataset of Images, create ZIP and CSV files.
9. Divide the dataset for Training and Testing (Figs. 1, 2).

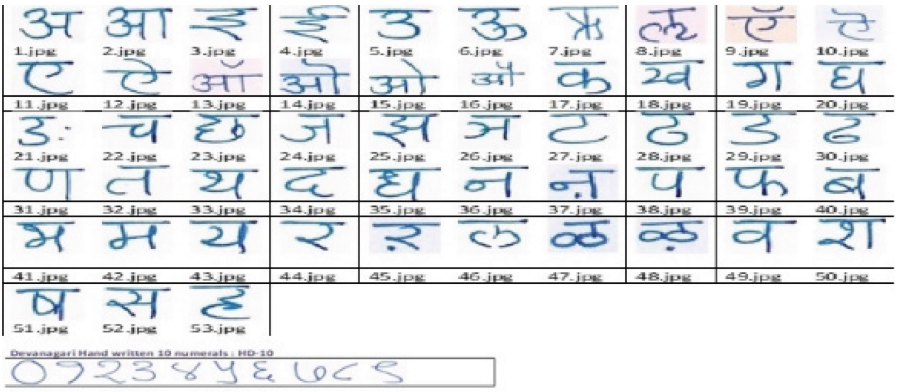


Fig. 2. Devanagari Characters Set

The Devanagari script comprises a total of 52 characters, includes 12 vowels, 10 numerals, and 31 consonants.

b. Rejected characters for design, with a justification:
We have omitted nineteen letters from the list that contain vowel modifiers, as these characters frequently result in user confusion and impede the computer's ability to accurately decipher CAPTCHA images (Fig. 3).

c. chosen character set:
Of the 52 Devanagari alphabets, 34 have been selected, along with ten numbers (Fig. 4).

d. **Design Canvas:**
The use of Python allows for the creation of a canvas, which enables the user to designate a character and employ the mouse to delineate the character on the canvas. The imagery produced by the user is saved in the relevant folder with a preset size of 300 x 300 pixels (Fig. 5).



Fig. 3. Rejected Devanagari Characters

Numerals (10)	Vowels (04)	Consonants (30)
० १ २ ३ ४ ५ ६ ७ ८ ९	अ इ उ ए	क ख ग घ च छ ज झ ट ड ढ ण त थ द ध न प ब भ म य र ल व श ष स ह ळ

Fig. 4. Selected Devanagari Character Set 10 Numerals, 4 Vowels, 30 Consonanats



Figure V : CANVAS User Interface

Fig. 5. Designed user interface model using Python

e. Information Storage:
Figure 6 accurately depicts the storage directory structure for the dataset. To guarantee efficient and hassle-free access, it is crucial to create directories and files and store images in their respective locations. The dataset is organized through the utilization

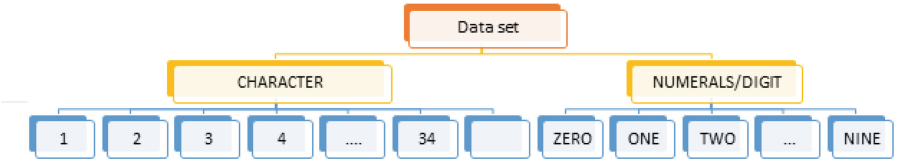


Fig. 6. Directory/Folder Strucre for Data storage

of two separate folders: Character and Digit. The Character folder comprises a total of 34 individual folders, each containing images of Devanagari characters. Additionally, there are ten folders dedicated to images of numerals.

f. **Pre-processing:**
The image is converted to grayscale, resized from 300 by 300 pixels to 65 by 65 pixels, stored in the (dot.jpeg format) and free of noise, ensuring each image occupies 1.5 kilobytes of storage space, and less pre-processing is required.

4.2 The Handwritten Devanagari Character Recognition Technique Uses Deep Learning

The dataset comprises 44,000 images, each depicting one of the foty four Devanagari alphabets. Each image measures 65 by 65 pixels and is assigned a class label 0 to 43. An illustration of these images alongside their respective labels is presented in Table 3 (Fig. 7).

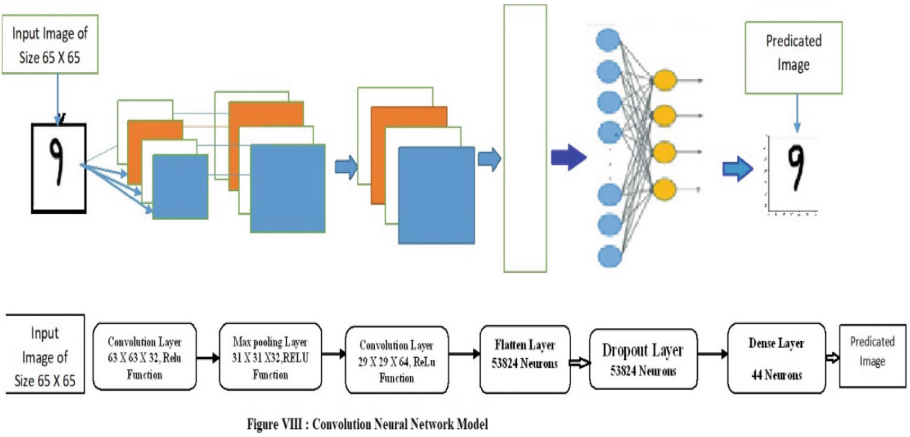


Fig. 7. Convolution Neural Network model for Devanagari Character recognition

To extract features, a Convolutional Neural Network (CNN) was employed for recognizing Devanagari alphabets and digits in images. In CNNs, each convolution layer incorporates filters that are shared among multiple neurons within the layer. These filters, smaller than the image size, are used to extract features from the input image. The portion of the image from which the filter extracts features is termed the receptive field, and the

extracted feature is termed a feature map. The dot product of the filter and the previous layer is stored in separate neurons of the convolution layer [20]. Implementation was carried out using Keras' Sequential model, composed of the following layers:

1. Convolutional Layer 2. Pooling layer 3. The Conv2D layer with ReLU activation function 4. Flatten layer 5. Dropout layer 6. Dense layer

1. Binary step 2. Linear 3. ReLU 4. Leaky ReLU 5. Sigmoid 6. Tanh 7. Softmax

1. **Binary Step Activation Function:** This activation function is used if we want to bound output. It is basically a threshold base classifier, in this, we decide some threshold value to decide output that neuron should be activated or deactivated. $f(x) = 1 \text{ if } x > 0 \text{ else } 0 \text{ if } x < 0$

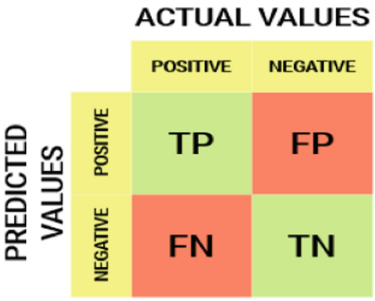
   Here the threshold value is 0. It is very simple and useful to classify binary problems or classifier.
2. **Linear Activation Function:** It is a simple straight line activation function where our function is directly proportional to the weighted sum of neurons or input. **Linear activation functions** are better in giving a wide range of activations and a line of a positive slope may increase the firing rate as the input rate increases. $Y = mZ$. Where derivative with respect to Z is constant m. The meaning gradient is also constant and it has nothing to do with Z.
3. **ReLU (Rectified Linear unit) Activation function:** Rectified linear unit or ReLU is most widely used activation function which ranges from **0 to infinity**, All the negative values are converted into zero, and this conversion rate is so fast that neither it can map nor fit into data properly. We use Leaky ReLU function instead of ReLU to avoid this unfitting, in Leaky ReLU range is expanded which enhances the performance.
4. **Leaky ReLU Activation Function:** We needed the **Leaky ReLU activation** function to solve the '***Dying ReLU***' problem, here we do not make all negative inputs to zero but to a value near to zero which solves the major issue of ReLU activation function.
5. **Sigmoid Activation Function:** It is basically is a probabilistic approach towards decision making and ranges in between **0 to 1,** so when we have to make a decision or to predict an output, we use this activation function because of the range is the minimum, therefore, prediction would be more accurate. The equation for the sigmoid function is $f(x) = 1/(1 + e^{(-x)})$

   The sigmoid function causes a problem mainly termed as vanishing gradient problem which occurs because we convert large input in between the range of 0 to 1 and therefore their derivatives become much smaller which does not give satisfactory output. To solve this problem another activation function such as ReLU is used where we do not have a small derivative problem.
6. **Hyperbolic Tangent Activation Function (Tanh):** This is also used to predict or to differentiate between two classes but it maps the negative input into negative quantity only and ranges in between -1 **to 1.**
7. **Softmax Activation Function:** Softmax is used mainly at the last layer i.e. output layer for decision making the same as sigmoid activation works, the softmax basically gives value to the input variable according to their weight and the sum of these weights is eventually one.For **Binary classification**, both **sigmoid**, as well as **softmax**, are equally approachable but in case of multi-class classification problem we generally use softmax and cross-entropy along with it.

There are 2,387,116 trainable parameters in this model overall.

h. Confusion Matrix:

A confusion matrix is organized as follows:



True Positive (TP): The number of instances that are predicted as positive and positive.
False Positive (FP): The number of instances that are predicted as positive but are negative.
False Negative (FN): The number of instances that are predicted as negative but are positive.
True Negative (TN): The number of instances that are predicted as negative and are negative.

The confusion matrix can be used to calculate various evaluation metrics for the model, such as accuracy, precision, recall, F1 score, and others.

Precision Score: Model precision score represents the model's ability to correctly predict the positives out of all the positive prediction it made.

Precision score is a useful measure of success of prediction when the classes are very imbalanced.

Mathematically, it represents the ratio of true positive to the sum of true positive and false positive.

$$\textbf{Precision Score} = \textbf{TP} / (\textbf{FP} + \textbf{TP})$$

Recall Score: Model recall score represents the model's ability to correctly predict the positives out of actual positives. This is unlike precision which measures as to how many predictions made by models are actually positive out of all positive predictions made.

Recall score is a useful measure of success of prediction when the classes are very imbalanced.

Mathematically, it represents the ratio of true positive to the sum of true positive and false negative.

$$\textbf{Recall Score} = \textbf{TP} / (\textbf{FN} + \textbf{TP})$$

Model accuracy score represents the model's ability to correctly predict both the positives and negatives out of all the predictions.

Mathematically, it represents the ratio of sum of true positive and true negatives out of all the predictions.

$$\textbf{Accuracy Score} = (\textbf{TP} + \textbf{TN})/(\textbf{TP} + \textbf{FN} + \textbf{TN} + \textbf{FP})$$

Model F1 score represents the model score as a function of precision and recall score. This is useful measure of the model in the scenarios where one tries to optimize either of precision or recall score and as a result, the model performance suffers.

The following represents the aspects relating to issues with optimizing either precision or recall score:

Optimizing for recall helps with minimizing the chance of not detecting a malignant cancer. However, this comes at the cost of predicting malignant cancer in patients although the patients are healthy (a high number of FP).

Optimize for precision helps with correctness if the patient has a malignant cancer. However, this comes at the cost of missing malignant cancer more frequently (a high number of FN).

Mathematically, it can be represented as harmonic mean of precision and recall score.

$$\textbf{F1 Score} = 2 * \textbf{Precision Score} * \textbf{Recall Score}/(\textbf{Precision Score} + \textbf{Recall Score}/)$$

5 Proposed Model

5.1 Environment for Experiments

This project is implemented on the Jupyter platform, utilizing the Windows environment and the Python language, version 3.0.0 dated February 20, 2020.

The required computer hardware specifications include an Intel(R), Core™, i5 or newer processor, CPU @ 2.20 GHz, 8 GB of RAM, 4 GB NVIDIA GEFORCE GTX GPU, and a 64-bit Windows operating system.

The project utilizes various Python libraries, including NumPy, Pandas, Scikit-learn (Sklearn), TensorFlow, Keras, OpenCV, PyGame, PyTorch, and Tesseract OCR. The OpenCV library employs the Morphing technique, which involves merging different images through a smooth transition to create a new image. OpenCV functions are used to read, write, display, resize, translate, scale, and rotate images.

5.2 Algorithm for Dataset Design

a. A comprehensive selection process was employed to identify 100 + native Devanagari speakers who were also proficient in computer usage, encompassing students from high schools, colleges, service industry professionals, and businesspeople of various age groups. The selected individuals were based in locations such as Jalgaon, Bhusawal, Chalisgaon, Dhule, Shirpur, Mumbai, Pune, Nashik, all located within the state of Maharashtra in India.
b. A Python-based canvas code was created and subsequently distributed to all 100 + users via email.

c. Each user was required to install Python on their device and execute the provided canvas code.
d. All users were trained using online meetings to draw Devanagari numerals, vowels, and consonants using a mouse.
e. The users collaboratively created 440 images using the canvas by drawing each of the forty-four characters ten times and saving them to their individual computers. The size of each character image was eight kilobytes, with a resolution of 300 by 300 pixels. Additionally, each user sent 440 JPEG images via email.
f. A total of 44,000 digitally rendered images were collected and manually arranged in respective folders, with each user contributing 440 images (44 characters x 10 repetitions) and a total of 100 users participating in the process.
g. Written Python code was developed for various tasks, including:
   i. Converting the image from RGB to grayscale.
   ii. Converting the image from RGB to grayscale.
   iii. Removing noise from the image.
   iv. Converting each image to JPEG format.
h. Each digital image spans 65 pixels in width and 65 pixels in height, while occupying a storage capacity of 1.5 kilobytes. Since they exist in a digital format, these images are free of any noise or the need for preprocessing.
i. A zip file and CSV file containing all images were created.

Resizing the image from 300 by 300 pixels to 65 by 65 pixels.

5.3 Devanagari Charatcer Recognition Using Convolutional Neural Network (CNN) Algorithm

a. Designated numerical identifiers spanning from 0 to 43 were assigned to the Devanagari digits and alphabet symbols. A curated selection of these images, along with their respective class labels, is showcased in Table 1. The process of inputting data into each layer and obtaining corresponding outputs can be scrutinized within the CNN framework.
b. We trained a total of 900 images for each class, resulting in a grand total of 39,600 alphabet images being trained.
c. We conducted tests on 4,400 images, which comprised of 100 images for each class, using the aforementioned CNN model across 10, 20, and 30 epochs.
d. The implementation of the Python code for character recognition through the CNN algorithm has been completed within a Python Jupyter Notebook 6.3.0 (Tables 2).

Table 1. Class Label Classification

Sample Image	Devanagari Character	Class label assigned	Sample Image	Devanagari Character	Class label assigned	Sample Image	Devanagari Character	Class label assigned
	०	**0**		ग	**16**		प	**31**
	१	**1**		घ	**17**		ब	**32**
	२	**2**		च	**18**		भ	**33**
	३	**3**		छ	**19**		म	**34**
	४	**4**		ज	**20**		य	**35**
	५	**5**		झ	**21**		र	**36**
	६	**6**		ट	**22**		ल	**37**
	७	**7**		ड	**23**		ळ	**38**
	८	**8**		ढ	**24**		व	**39**
	९	**9**		ण	**25**		श	**40**
	अ	**10**		त	**26**		ष	**41**
	इ	**11**		थ	**27**		स	**42**
	उ	**12**		द	**28**		ह	**43**
	ए	**13**		ध	**29**			
	क	**14**		न	**30**			

5.4 Python Code Executed Shows Layer-Wise Output Its Interpretation Is

Table 2. Layer-wise output interpretation

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 63, 63, 32)        320

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 29, 29, 64)        18496

 flatten (Flatten)           (None, 53824)             0

 dropout (Dropout)           (None, 53824)             0

 dense (Dense)               (None, 44)                2368300

=================================================================
Total params: 2,387,116
Trainable params: 2,387,116
Non-trainable params: 0
_________________________________________________________________
```

Table 3. Recognition Percentage per Epochs

Epochs	Images tested	Recognized	Not Recognized	% Recognition	% Not Recognition
10	440	422	18	95.90	4.09
20	440	414	26	94.09	5.92
30	440	424	16	**96.36**	3.63

Comparative analysis of epochs-wise recognition of Devanagari alphabets (Figs. 8, 9):

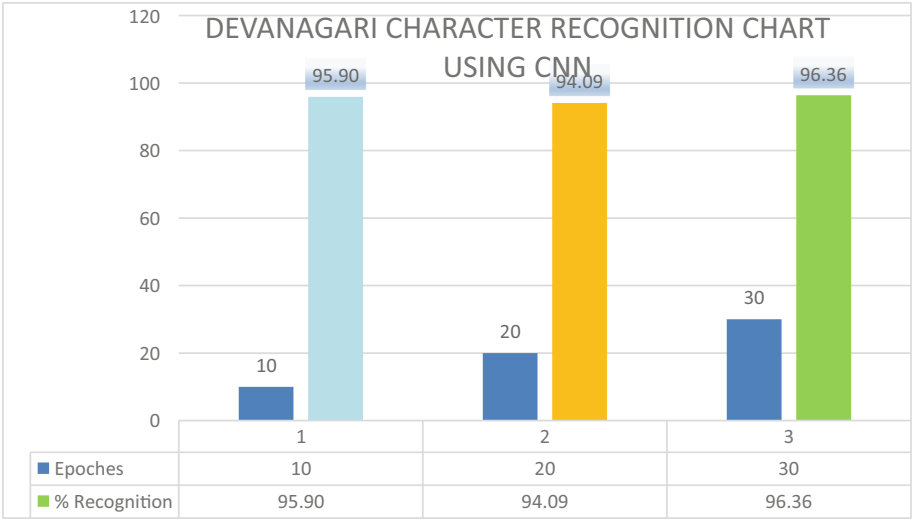


Chart. 1. Epoch-wise percentage of Devanagari Character recognition using CNN

5.5 CNN Model Accuracy and Loss

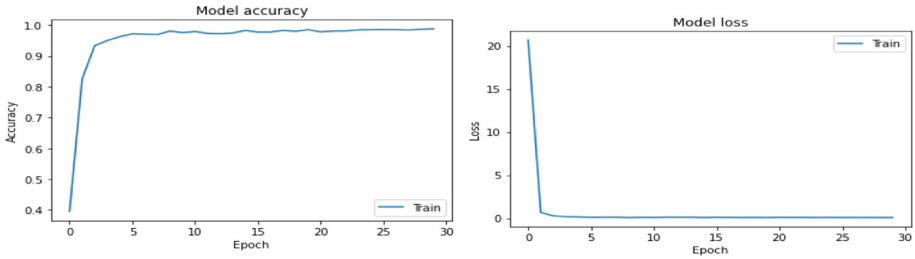


Chart. 2. Epoch-wise model accuracy and loss

5.6 Output of Devanagari Character Recognition by CNN Model

The output of Devanagari Character recognition by the CNN model is depicted in Tables 3, 4.

5.7 Database of CNN Model Output

Developed a database in Excel to store the CNN model's output. The database consists of a table that contains the input image, image number, predicted image, and corresponding accuracy. The table is illustrated in Table 5.

Based on the data in the table, we determined the outcome for each input image, with a result of '0' indicating a match with the predicted image and a result of '1' indicating

Table 4. Devanagari Character recognition Test Result using CNN Model

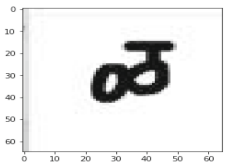
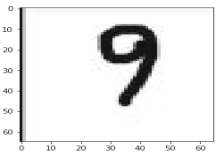
		
The character Number is 38 **Prediction is ळ** Accuracy : 1.0 100.0 File :274.jpeg	The character Number is 1 **Prediction is १** Accuracy : 1.0 100.0 File :132.jpeg	The charcater Number is 40 **Prediction is श** Accuracy : 0.99 100.0 File :470.jpeg

Table 5. Database of Predicated Images and their output

FOLDER	SYMBOL	IFILENO	SYMBOLNO	PSYMBOL	ACCURACY	RESULT	EPOCH
0	०	98	0	०	1	0	30
1	१	11	1	१	1	0	30
2	२	148	2	२	1	0	30
3	३	391	3	३	0.99	0	30
4	४	192	4	४	1	0	30
5	५	489	5	५	1	0	30
6	६	497	6	६	1	0	30
7	७	476	7	७	1	0	30
8	८	376	8	८	1	0	30
9	९	137	9	९	1	0	30
10	अ	133	10	अ	1	0	30
11	इ	309	11	इ	1	0	30
12	उ	51	12	उ	1	0	30
13	ए	150	13	ए	0.971726	0	30
14	क	261	14	क	1	0	30
15	ख	164	15	ख	1	0	30

a misidentification. We then identified all rows where the result was ‘1’, which signifies that the model incorrectly recognized an input image that was not part of the set of known images.

5.8 Number of Times Image Recognized and Not Recognized Table

A new table has been created that presents a class-wise and image-wise summary of the data from the previous table. This table includes attributes such as class, Devnagari alphabet (D-ALPHABET), the number of times the image was correctly predicted (RECOG.), and the number of times the image was incorrectly predicted (NR). The data presented in Table V demonstrates the frequency of inaccurate predictions or interpretations of images performed by the model (Table 6).

The Devanagari numeral "0" is recognized properly 95 times and incorrectly 5 times, the model recognizes Devanagari digit '0' as Devanagari digit '७' three times and Devanagari digit '६' as two times.

In the aforementioned table entry, the letter combination "gha" घwas correctly identified 94 times, but incorrectly predicted six times. Of these, it was mistakenly predicted as Devanagari consonant 'ध' 'dha', two times 'य' 'ya', and one time kha ख. ("dh" (3 times), "ya" (2 times), and "kh" once.)

The symbols or images: **उ,च,ज,ण, प, ष**("u," "ch," "j," "n," "p," and "s") were successfully recognized at 100%.

5.9 The Confusion Matrix

Based on the information in Table 5, the data has been entered into the confusion matrix.

a. The diagonal values indicate that character have been appropriately recognized.
b. The values in the other rows represent the incorrect recognition of input character image, which are added to the corresponding column. The row sum must be 100, since we have only tested 100 images.

The top row of the confusion matrix reflects that the image/digit '0' has been accurately identified 95 times out of 100 attempts, and incorrectly identified 5 times

A summary of the Confusion Matrix parameters can be found in Table 7, which is utilized to construct Table 8. The following attributes are present in Table 8:

Table 6. Alphabet Test Result [Recog.(Recognized), NR (Not Recognized)] using CNN Model

CLASS	S	NR	R	WRONGLY -RECOGNISED SYMBOLS													
0	०	5	95	६	७	७	६	७									
1	१	13	87	७	७	७	भ	र	७	ज	७	ज	७	र	७	र	
2	२	11	89	१	३	३	३	क	क	क	क	क	क	३			
3	३	13	87	उ	इ	८	उ	८	०	०	उ	०	८	८	८	इ	
4	४	1	99	९													
5	५	1	99	१													
6	६	12	88	ढ	ड	इ	ड	४	४	ड	५	इ	इ	४	२		
7	७	3	97	ख	ख	ख											
8	८	2	98	४	४												
9	९	8	92	६	४	८	४	४	४	४	६						
10	अ	7	93	५	७	५	५	थ	थ	थ							
11	इ	2	98	व	व												
12	**उ**	**0**	**100**														
13	ए	3	97	ष	प	प											
14	क	3	97	ळ	६	६											
15	ख	3	97	भ	३	३											
16	ग	5	95	ळ	ढ	ढ	ळ	ढ									
17	घ	6	94	ख	ध	ध	ध	य	य								
18	**च**	**0**	**100**														
19	छ	5	95	ज	ज	ध	ज	ज									
20	**ज**	**0**	**100**														
21	झ	2	98	९	९												
22	ट	2	98	ढ	ढ												
23	ड	3	97	व	६	६											
24	ढ	3	97	द	द	ळ											
25	**ण**	**0**	**100**														
26	त	8	92	ल	ल	ल	ल	ल	ल	ल	ब						
27	थ	3	97	भ	२	२											
28	द	4	96	झ	झ	ह	६										
29	ध	2	98	ल	ल												
30	न	3	97	त	त	त											
31	**प**	**0**	**100**														
32	ब	3	97	भ	भ	थ											

(*continued*)

Table 6. (*continued*)

33	भ	11	89	उ	थ	अ	७	थ	अ	अ	७	४	४	उ		
34	म	10	90	ढ	ब	भ	ह	ह	ह	ब	भ	ब	ढ			
35	य	4	96	झ	थ	झ	झ									
36	र	1	99	प												
37	ल	8	92	६	ब	ब	थ	ब	थ	६	ब					
38	ळ	2	98	अ	ब											
39	व	8	92	ब	छ	छ	न	ब	छ	छ	ब					
40	श	4	96	ल	भ	भ	भ									
41	**ष**	**0**	**100**													
42	स	11	89	ष	ष	झ	ष	झ	झ	झ	श	श	झ	झ		
43	ह	6	94	द	द	ध	ड	ड	ड							

S: Symbol R: Recognized NR: Not-Recognized

True Positive refers to an instance wherein the predicted class label aligns accurately with the actual class label in the diagonal element of both the row and column.

False Negative, conversely, is the sum of all values in the row, with the exception of the diagonal element. False Positive refers to the sum of all values in the column, excluding the diagonal element.Finally, True Negative is the sum of all elements in the confusion matrix (4400), subtracting the sum of false positives and false negatives (FP + FN) [10].

Subsequently, we assess the performance analysis parameters—Precision, Recall, F1-score, Accuracy, and Specificity—utilizing the respective values for each class.

According to the aforementioned table, 4,199 alphabet images are correctly recognized, while only 201 images are not recognized out of the total of 4,400 alphabet images. The recognition rate constitutes 95.44%, while the non-recognition rate amounts to 4.58%.

5.10 Performance Evaluation Criteria

I. Accuracy: The accuracy is the proportion of correct predictions made by the model, calculated as (True Positive + True Negative) / (True Positive + True Negative + False Potive + False Negative).
II. Precision: Precision is the proportion of true positives among the predicted positives, calculated as True Positive / (True Positive + False Potive).
III. Recall (Sensitivity): Recall is the proportion of true positives among the actual positives, calculated as True Positive / (True Positive + False Negative).
IV. Specificity: Specificity is the proportion of true negatives among the actual negatives, calculated as True Negative / (True Negative + False Potive).

Table 7. Confusion Matrix for the 100 Test Images

A	CL	०	१	२	३	४	५	६	७	८	९	अ	इ	उ	ए	क	ख	ग	घ	च	छ	ज	झ	ट	ड	ढ	ण	त	थ	द	ध	न	प	ब	भ	म	य	र	ल	ळ	व	श	ष	स	ह	
०	0	95	0	0	0	0	0	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
१	1	0	97	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	100
२	2	0	1	89	4	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
३	3	3	0	0	87	0	0	0	0	5	0	0	2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
४	4	0	0	0	0	99	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
५	5	0	1	0	0	0	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
६	6	0	0	1	0	3	1	88	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
७	7	0	0	0	0	0	0	0	97	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
८	8	0	0	0	0	2	0	0	0	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
९	9	0	0	0	0	5	0	2	0	1	92	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
अ	10	0	0	0	0	0	0	3	1	0	0	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
इ	11	0	0	0	0	0	0	0	0	0	0	0	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	100
उ	12	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
ए	13	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	0	0	100
क	14	0	0	0	0	0	0	2	0	0	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	100
ख	15	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	100
ग	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	95	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	100
घ	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	94	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	2	0	0	0	0	0	0	0	0	100
च	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
छ	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	95	4	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	100
ज	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
झ	21	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
ट	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
ड	23	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	100
ढ	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	100
ण	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
त	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	92	0	0	0	0	0	1	0	0	0	0	7	0	0	0	0	0	100
थ	27	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	100
द	28	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	96	0	0	0	0	0	0	0	0	0	0	0	0	1	100
ध	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98	0	0	0	0	0	2	0	0	0	0	0	0	100
न	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	100
प	31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	100
ब	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	97	0	2	0	0	0	0	0	0	0	0	100
भ	33	0	0	0	0	2	0	0	2	0	0	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	89	0	0	0	0	0	0	0	0	0	100
म	34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	3	2	90	0	0	0	0	0	0	0	3	100
य	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	0	0	96	0	0	0	0	0	0	0	100
र	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	99	0	0	0	0	0	0	100
ल	37	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	4	0	0	0	0	92	0	0	0	0	0	100
ळ	38	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	98	0	0	0	0	100
व	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	0	92	0	0	0	100
श	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	0	0	96	0	0	100
ष	41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	100
स	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	89	0	100
ह	43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	94	100
		०	१	२	३	४	५	६	७	८	९	अ	इ	उ	ए	क	ख	ग	घ	च	छ	ज	झ	ट	ड	ढ	ण	त	थ	द	ध	न	प	ब	भ	म	य	र	ल	ळ	व	श	ष	स	ह	

V. F1-Score: The F1-Score is the harmonic mean of precision and recall, calculated as 2 * (precision * recall) / (precision + recall). It is presented in the performance evaluation table below [10].

Table 8. Calculation of TP, FN, TN, FP

Symbol	TP	FN	TN	FP
०	95	5	4392	3
१	87	13	4385	2
२	89	11	4386	3
३	87	13	4381	6
४	99	1	4387	12
५	99	1	4398	1
६	88	12	4374	14
७	97	3	4384	13
८	98	2	4392	6
९	92	8	4389	3
अ	93	7	4389	4
इ	98	2	4393	5
उ	100	0	4395	5
ए	97	3	4397	0
क	97	3	4391	6
ख	97	3	4393	4
ग	95	5	4395	0
घ	94	6	4394	0
च	100	0	4400	0
छ	95	5	4391	4
ज	100	0	4394	6
झ	98	2	4387	11
ट	98	2	4398	0
ड	97	3	4391	6
ढ	97	3	4389	8

(*continued*)

Table 8. (*continued*)

ण	100	0	4400	0
त	92	8	4389	3
थ	97	3	4388	9
द	96	4	4392	4
ध	98	2	4393	5
न	97	3	4396	1
प	100	0	4397	3
ब	97	3	4385	12
भ	89	11	4381	8
म	90	10	4388	2
य	96	4	4394	2
र	99	1	4396	3
ल	92	8	4382	10
ळ	98	2	4394	4
व	92	8	4389	3
श	96	4	4394	2
ष	100	0	4396	4
स	89	11	4389	0
ह	94	6	4390	4
Total	4199	201	193198	201

6 Result Analysis

6.1 Devanagari Dataset Design

Samples of Dataset 44000, 65 by 65 pixels vowels, consonants, and numerals images (Fig. 10):



Fig. 10. Devanagari Dataset Samples

Creation of Datasets and CSV Files

A CSV (comma-separated values) file, which is a type of text file that follows a specific format, is commonly used to store data in a table-like structure that is compatible with a variety of tools. These files are especially useful when it is necessary for data to be compatible with a wide range of tools. They can be opened by text editors and specialized software, as well as spreadsheet programs like Excel. In this particular case, the images of each character and digit are saved in a CSV (Comma Separated Values) spreadsheet file, with each image being 65 pixels by 65 pixels in size. To create a grayscale version of an RGB image, Python code is used to extract the pixel values. The pixel values for each image are then recorded in a single row of the spreadsheet, with a total of 4225 pixels (ranging from 0 to FFN). Separate CSV files are created for each character and digit, with one file designated for training and the other for testing. The images are then

partitioned, with 70% being allocated for the training dataset folder and 30% for the testing dataset folder.

Dataset Publication: The purpose of publishing this dataset is to facilitate recognition and other applications.

1. Mendeley Data: This dataset is available on Mendeley Data, with a DOI of https://doi.org/10.17632/yb9rmfjzc2.1, dated 5 Oct. 2022 [8].
2. IEEE DataPort: The dataset is also available on IEEE DataPort, with a DOI of https://doi.org/10.21227/9zpv-3194, dated 6 Oct. 2022 [9].

6.2 CNN-Assisted Handwritten Devanagari Character Recognition

Evaluation Parameters

According to the information presented in Table 9 and the evaluation parameters set forth, it was concluded that an effective classification model should display low false positive and false negative rates, in addition to demonstrating high levels of accuracy, precision, recall, specificity, and F1-score.

The statistics for the evaluation metrics utilized for the CNN model are displayed in Table 10.

The overwhelming majority of accuracy scores are remarkably high, with an abundance of them surpassing 0.99 and even reaching 1. These exceptional scores testify to the classifier's proficiency in discerning patterns within the data and producing precise predictions for each class. It is noteworthy that a multi-class accuracy score of more than 0.99 is considered extraordinary, and a high level of accuracy is typically pursued in any classification task (Fig. 11).

Table 9. Performance Evaluation matrix

Symbol	Diagonal	Row	EXCEPT-R-C	Column	TP/ TP+FP	TP/ TP+FN	2*P*R/ (P+R)	(TP+TN) /(TP+TN +FP+FN)	TN/ (TN+FP)
	TP	FN	TN	FP	Precision (P)	Recall (R)	F1-score	Accuracy	Specific-ity
०	95	5	4392	3	0.9694	0.9500	0.9596	0.9982	0.9993
१	87	13	4385	2	0.9775	0.8700	0.9206	0.9967	0.9995
२	89	11	4386	3	0.9674	0.8900	0.9271	0.9969	0.9993
३	87	13	4381	6	0.9355	0.8700	0.9016	0.9958	0.9986
४	99	1	4387	12	0.8919	0.9900	0.9384	0.9971	0.9973
५	99	1	4398	1	0.9900	0.9900	0.9900	0.9996	0.9998
६	88	12	4374	14	0.8627	0.8800	0.8713	0.9942	0.9968
७	97	3	4384	13	0.8818	0.9700	0.9238	0.9964	0.9970
८	98	2	4392	6	0.9423	0.9800	0.9608	0.9982	0.9986
९	2	8	4389	3	0.9684	0.9200	0.9436	0.9976	0.9993
अ	93	7	4389	4	0.9588	0.9300	0.9442	0.9976	0.9991
इ	98	2	4393	5	0.9515	0.9800	0.9655	0.9984	0.9989
उ	100	0	4395	5	0.9524	1.0000	0.9756	0.9989	0.9989
ए	97	3	4397	0	1.0000	0.9700	0.9848	0.9993	1.0000
क	97	3	4391	6	0.9417	0.9700	0.9557	0.9980	0.9986
ख	97	3	4393	4	0.9604	0.9700	0.9652	0.9984	0.9991
ग	95	5	4395	0	1.0000	0.9500	0.9744	0.9989	1.0000
घ	94	6	4394	0	1.0000	0.9400	0.9691	0.9987	1.0000
च	100	0	4400	0	1.0000	1.0000	1.0000	1.0000	1.0000
छ	95	5	4391	4	0.9596	0.9500	0.9548	0.9980	0.9991
ज	100	0	4394	6	0.9434	1.0000	0.9709	0.9987	0.9986
झ	98	2	4387	11	0.8991	0.9800	0.9378	0.9971	0.9975
ट	98	2	4398	0	1.0000	0.9800	0.9899	0.9996	1.0000
ड	97	3	4391	6	0.9417	0.9700	0.9557	0.9980	0.9986
ढ	97	3	4389	8	0.9238	0.9700	0.9463	0.9976	0.9982
ण	100	0	4400	0	1.0000	1.0000	1.0000	1.0000	1.0000
त	92	8	4389	3	0.9684	0.9200	0.9436	0.9976	0.9993
थ	97	3	4388	9	0.9151	0.9700	0.9417	0.9973	0.9980
द	96	4	4392	4	0.9600	0.9600	0.9600	0.9982	0.9991
ध	98	2	4393	5	0.9515	0.9800	0.9655	0.9984	0.9989

(*continued*)

Table 9. (*continued*)

न	97	3	4396	1	0.9898	0.9700	0.9798	0.9991	0.9998
प	100	0	4397	3	0.9709	1.0000	0.9852	0.9993	0.9993
ब	97	3	4385	12	0.8899	0.9700	0.9282	0.9967	0.9973
भ	89	11	4381	8	0.9175	0.8900	0.9036	0.9958	0.9982
म	90	10	4388	2	0.9783	0.9000	0.9375	0.9973	0.9995
य	96	4	4394	2	0.9796	0.9600	0.9697	0.9987	0.9995
र	99	1	4396	3	0.9706	0.9900	0.9802	0.9991	0.9993
ल	92	8	4382	10	0.9020	0.9200	0.9109	0.9960	0.9977
ळ	98	2	4394	4	0.9608	0.9800	0.9703	0.9987	0.9991
व	92	8	4389	3	0.9684	0.9200	0.9436	0.9976	0.9993
श	96	4	4394	2	0.9796	0.9600	0.9697	0.9987	0.9995
ष	100	0	4396	4	0.9615	1.0000	0.9804	0.9991	0.9991
स	89	11	4389	0	1.0000	0.8900	0.9418	0.9975	1.0000
ह	94	6	4390	4	0.9592	0.9400	0.9495	0.9978	0.9991
T	**4199**	**201**	**193198**	**201**	**0.9555**	**0.9543**	**0.9543**	**0.9980**	**0.9990**

Table 10. Assessment Factors

Assessment Factors	Range	Average
Precision	0.86 to 1.0	0.956
Recall	0.87 to 1.0	0.954
F1-Score	0.87 to 1.0	0.954
Accuracy	0.99 to 1.0	0.998
Specificity	0.99 to 1.0	0.990

7 Comparative Analysis

7.1 Dataset Design

Studied the most pertinent research papers related to the literature review issue. Table 11 currently shows a comparative analysis of all review papers. It is discovered that while our dataset employs a novel online handwriting technique, all other datasets are created using offline techniques. While pre-processing was necessary for all datasets, it was not necessary for our dataset. Produced a dataset with 100% less noise. [8, 9].

7.2 Dataset Recognition

Acknowledged Our specially created dataset has a 99% accuracy rate, which is higher than that of other research studies (Table 12).

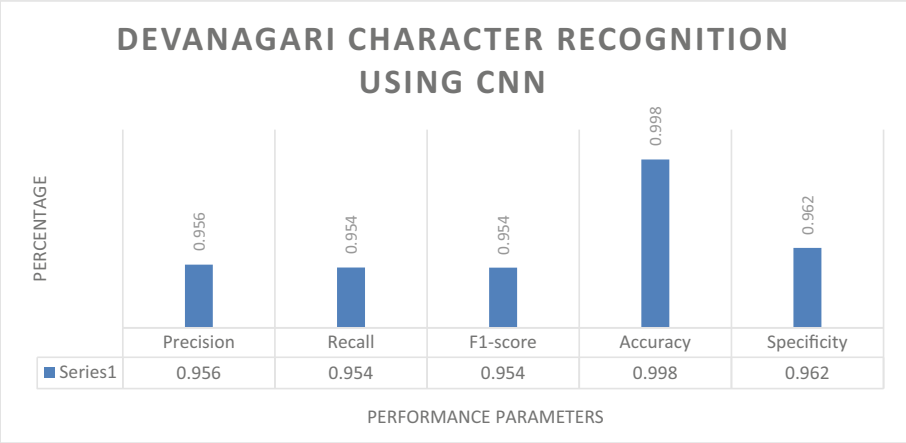


Chart. 3. Devanagari Character Recognition Using CNN

Table 11. Comparative Analysis of Dataset Design research papers

Pub Year	Character set				Total Images in Dataset	Data Format	Image Size (Pixel)	Repository Name	Data Acquisition
	Digit	Vowels	Consonant	Total					
2009 [2]	10	12	36	58	22,556	JPEG	80 x 80	Available open source	Handwritten, scanned, and Processed images
2012 [3]	10	12	36	58	12,912	JPEG, CSV, ZIP	28 X 28	Kaggle Dataset	Collected from 40 users
2014 [4]	14-M	13	36	58	35,000 83,300 1,18,300	JPG	28 X 28	Available Open Source	Collected from Native users
2016 [5]	10	--	36	46	92,000	PNG, CSV, ZIP	32 X 32	The University of California, Irvine (UCI) Machine Learning Repository	Created by extraction and manual annotation of thousands of characters from handwritten documents

(*continued*)

Table 11. (*continued*)

Pub Year	Character set				Total Images in Dataset	Data Format	Image Size (Pixel)	Repository Name	Data Acquisition
	Digit	Vowels	Consonant	Total					
2021 [6]	10	12	36	58	5,800	JPG	1600 X 1600	Kaggle Dataset 10.1007/s12046-020-01484-1	Handwritten characters using ball pen from local writers
2021 [7]	10	13	--	23	38,750	JPEG, CSV	28 X 28 Pixels	Mendeley Data 10.17632/pxrnvp4yy8.3	Collected on plain A4 sheets, scanned using Epson – 150
2022 [8]	10	4	30	44	44,000	JPEG	65 X 65	Mendeley	Online Digital using UI – Canvas
2022 [9]	10	4	30	44	44,000	JPEG	65 X 65	IEEE data port	Online Digital using UI – Canvas

Table 12. Comparative Analysis of Devanagari Character recognition Papers

Reference	Classes	Dataset	Architecture	Accuracy
Narang et al. (2021)[17]	33	5484 Images	3 layers of CNN	93.73%
Dokare et al. (2021)[18]	48	92000 images	CNN with a dropout layer	97.56%
Bisht et al. (2021)[19]	--	8620 images	Double CNN	95.97%
Shalaka Deore (2020) [6]	58	5800 isolated images	Deep Convolutional Neural Network	96.55%
Prof.Rakesh Ramteke, Sanjay Pate [8, 9]	**44**	**Own dataset of 44,000 Isolated images**	**6 layers of Convolutional Neural Network**	**99.00%**

8 Conclusion and Future Work

The central aim of this research endeavor is to create a digital Handwritten Dataset containing 44,000 images of Devanagari script letters and numbers in an digital format. The dataset ought to be of exceptional quality, with a multi-class accuracy score of not less than 0.99. The researchers envision expanding the dataset to incorporate the remaining 19 Devanagari symbols, which would bolster the recognition model's generalization abilities. Additionally, the development of a dataset comprising alphabets and numerals

for other Indian languages is currently being contemplated. During the dataset creation process, users primarily utilized a mouse, which might have restricted the diversity of writing styles captured. Therefore, future research could explore different input methods or incorporate a broader range of writing tools to better represent the various writing styles. The research also explores the performance of advanced architectures in Devanagari character recognition and the potential for expanding the scope to include other Indic scripts or languages.

Future research should focus on optimizing algorithms and architectures for real-time processing in document analysis and interactive learning tools, and exploring the model's cross-script applicability, as generalization and adaptation may require further investigation and adaptation.

Revenue Acknowledgment. The authors did not receive any money to create the dataset that is included in this study.

Accessible Dataset The following is a public dataset of our recently generated Digital Handwritten Devanagari characters:

1. Mendeley Data, accessed at < https://data.mendeley.com/datasets/yb9rmfjzc2 >, bears the https://doi.org/10.17632/yb9rmfjzc2.1 [8].
2. The IEEE data port, with its https://doi.org/10.21227/9zpv-3194 [9], also contains relevant information.

References

1. Devanagari Details of Devanagari script in Wikipedia https://en.wikipedia.org/wiki/Devanagari,2022. Accessed 25 June 2022
2. Bhattacharya, U., Chaudhuri, B.B.: Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. IEEE Trans. Pattern Anal. Mach. Intell. **31**(3), 444–457 (2009)
3. Pant, A.K.: https://www.kaggle.com/datasets/ashokpant/devanagari-char-acter-dataset
4. Kumar, R., Ravulakollu, K.K.: Handwritten devnagari digit recognition: benchmarking on new dataset. J. Theor. Appl. Inf. Technol. **60**(3) (2014)
5. Acharya, S., Pant, A.K., Gyawali, P.K.: Deep learning based large scale hand-written devanagari character recognition. In: Proceedings of the 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), pp. 121–126 (2015)
6. Deore, S.P., Pravin, A.: devanagari handwritten character recognition using fine-tuned deep convolutional neural network on trivial dataset. Sādhanā **45**, 243 (2020). https://doi.org/10.1007/s12046-020-01484-1
7. Prashanth, D.S., Mehta, R.V.K., Challa, N.P.: A multi-purpose dataset of Devanagari script comprising of isolated numerals and vowels. Data Brief **40**, 107723 (2022). https://doi.org/10.1016/j.dib.2021.107723
8. Ramteke, R., Pate,S.: Handwritten devanagari character data set of 44,000 images for Devanagari CAPTCHA generation. IEEE Data Port. https://doi.org/10.21227/9zpv-3194
9. Ramteke, R., Pate, S.: Handwritten devanagari character data set of 44,000 images for Devanagari captcha generation. Mendeley Data. https://doi.org/10.17632/b9rmfjzc2.1
10. Karrach, L., Pivarčiová, E.: Using a convolutional neural network for machine written character recognition. Res. Square (2023). https://doi.org/10.21203/rs.3.rs-2712626/v1

11. Gurav, Y., Bhagat, P., Sinha, S.: Devanagari handwritten character recognition using convolutional neural networks. In: Proceeding of the 2nd International Conference on Electrical, Communication and Computer Engineering (ICECCE), Istanbul, Turkey 978-1-7281-7116-6/20 (2020)
12. Acharya, S., Pant, A.K., Gyawali, P.K.: Deep learning based large scale handwritten devanagari character recognition. In: 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA) (2015)
13. Aneja, N., Aneja, S.: Transfer learning using CNN for hand-written devanagari character recognition. In: 1st International Conference on Advances in Information Technology, pp. 293−296. 978-1- 7281-3241-9/19/$31.00 © 2019. IEEE (2019)
14. Prashanth, D.S., Mehta, R.V., Sharma, N.: Classification of handwritten devanagari number – an analysis of pattern recognition tool using neural network and CNN. Int. Conf. Comput. Intell. Data Sci. (ICCIDS) (2019)
15. Yadav, S., Pandey, A., Aggarwal, P., Garg, R., Aggarwal, V.: Handwriting recognition using LSTM networks. Int. J. New Technol. Res. (IJNTR) **4**(3), ISSN:2454. 4116 (2018)
16. Padmaja, K.: Devanagari handwritten character recognition using deep learning. Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET) **10**(1), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.53 (2022)
17. Narang, S.R., Kumar, M., Jindal, M.K.: Deep net Devanagari: a deep learning model for Devanagari ancient character recognition. Multimedia Tools Appl. **80**(13), 20671–20686 (2021)
18. Dokare, I., et al.: Recognition of handwritten devanagari character using convolutional neural network. In: 2021 3rd International Conference on Signal Processing and Communication (ICPSC). IEEE (2021)
19. Bisht, M., Gupta, R.: Offline handwritten Devanagari modified character recognition using convolutional neural network. Sādhanā **46**(1), 1–4 (2021)
20. Alzubaidi, L., Zhang, J., Humaidi, A.J., et al.: Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data **8**, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8

Racism Detecting in Twitter Comments Using Metaheuristics Optimized Text Mining and Classification

Dobrivoje Dubljanin[1], Luka Jovanovic[2], Nebojsa Bacanin[2(✉)], Petar Spalevic[3], Branislav Radomirovic[2], Miodrag Zivkovic[2], and Angelina Njegus[2]

[1] Technical Faculty in Bor, University of Belgrade, Vojske Jugoslavije 12, 19210 Bor, Serbia
[2] Singidunum University, Danijelova 32, 11000 Belgrade, Serbia
nbacanin@singidunum.ac.rs
[3] Faculty of Technical Science, University of Pristina, Kneza Milosa 7, 40000, 610101 Kosovska Mitrovica, Serbia

Abstract. The rise of social media platforms has fundamentally transformed how individuals interact and communicate, but as it facilitated relations between various internet users, it also precipitated an alarming increase in the spreading of racist remarks and hate-fuelled incidents. Key factors contributing to the proliferation of this harmful behavior include, among many others, the anonymity of users, digital disinhibition, and the very nature of social media with its lack of laws and restrictions, through the historical patterns of discriminatory language use, linguistic cues, and contextual analysis. This research introduces a comprehensive framework that integrates artificial intelligence, advanced natural language processing tools, and machine learning classifiers to identify racist content. Leveraging these methodologies and incorporating preprocessing techniques to reduce noise and biases present in social media data leads to more accurate identification of racially objectionable content. The demonstrative outcomes show the efficacy of this approach in identifying and classifying racist remarks on Twitter, supporting automatic content moderation, and encouraging a more welcoming online community. An additional contribution of this work is the introduction of a modified metaheuristics optimization method that is applied to classifier optimization to ensure favorable outcomes. A comparative analysis is conducted on real-world data and the best-performing optimized classifiers demonstrate an accuracy of 0.923058.

Keywords: Racism · Metaheuristics Optimization · Text Mining · Variable Neighborhood Search · AdaBoost · Machine Learning

1 Introduction

As social media has become a social hub for uniting people, especially youngsters, among many others, it comes with various benefits and downsides. It is primarily responsible for developing and enriching global relations and communications. However, with its

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 217–231, 2025.
https://doi.org/10.1007/978-3-031-86296-0_16

infinite size and almost unimaginable capabilities, there is also a dark side in that it has created opportunities for displaying malicious behavior with little to no consequences [19]. Users (re)produce racism by engaging in seemingly unsuspicious activities like using GIFs and emojis [18]. Research on racist discourses and practices on social media is important yet difficult. Because of the nature of these platforms, like Twitter, users are free to express their opinions however they see fit.

One of the potential solutions to this problem is using text mining in combination with advanced artificial intelligence (AI) algorithms. These algorithms can be updated with little programmatic intervention, learning from observed data. There is an abundance of textual material available, but the text is unstructured data, and unstructured data needs to have some method of extracting meaningful information [13]. The goal of text mining is to uncover patterns and helpful hidden information in text.

Not everything is perfect when working with these uncharted territories, and along the way, a lot of challenges arise. One of them specifically is hyperparameter tuning. Hyperparameter tuning is the process of selecting the optimal set of hyperparameters for a machine-learning (ML) [11]. Hyperparameter tuning is often considered an NP-hard problem, and the best approach to addressing this issue is using metaheuristic algorithms [28], as they can be utilized to yield decent outcomes.

This study aims to explore the possibilities of dealing with racism and harassment and consequently help with a better experience for internet users. In this study, the algorithms used will include Term Frequency-Inverse Document Frequency (TF-IDF) [4] and will be used in correlation with the AdaBoost [9] classifier. As the potential of these algorithms is closely related to appropriate hyperparameter selection, variable neighborhood search (VNS) [25] algorithm will also be included to optimize performance. The aforementioned combination will be tested on a publicly available real-world dataset and provide scientific contributions that include, but are not limited to:

- A suggestion for a data-driven method of identifying racism
- An investigation on the classification of harassment using NLP approaches in conjunction with classifiers
- A proposal of a modified metaheuristic optimizer that has been tweaked to better suit the requirements of this study

The remaining sections of this document are organized as follows: In Sect. 2, the previous efforts are acknowledged, and a concise summary of the literature is presented, emphasizing the research gap and motivations. A thorough explanation of the newly adopted technique is provided in Sect. 3. In Sect. 4, an experimental setup will be utilized to test suggested approaches, and the findings are shown in Sect. 5. In Sect. 6, the work is concluded, and it also addresses some of the shortcomings and suggestions for further study.

2 Related Works

In recent years, there has been a surge of similar projects that utilize text processing, namely language models such as generative pre-trained transformers [17], bidirectional encoder representations from transformers [7]. All the aforementioned language models

use transformer architecture and are very adept at processing text, summarizing large amounts of text, and even summarizing the meanings behind text. In a test conducted in 2024. [16], it was proven that there is a difference between these models in correlation with computational efficiency and effectiveness.

It takes a lot of effort and time to construct an ML model that works. The right algorithm needs to be chosen, and the model's hyper-parameters need to be adjusted to get the best possible model architecture [32]. Setting and finely tuning hyperparameters is essential in applied AI. This is one of the representatives of the NP-hard problem, and its solution is almost never simple, and every approach has various results. According to the No Free Lunch (NFL) [2] theorem, without resampling, the performance of all optimization techniques is the same when averaged over all optimization tasks. That means that there is no single best-suited optimizer for all problems, so several divergent metaheuristics have had to be developed.

Some notable examples of optimizers include algorithms drawing inspiration from nature such as the genetic algorithm (GA) [20], whale optimization algorithm (WOA) [23] and the particle swarm optimizer (PSO) [14]. More recent examples include the (SSA) [22], (RSA) [1] and (ChOA) [15] optimizer that utilized more advanced search strategies to handle optimization. Finally, more abstract inspirations have served as a basis for emerging optimizers such as the sinh cosh algorithms (SCHA) [3] that draw inspiration on somewhat abstract mathematical concepts to handle optimization in an effective manner. Several examples of real-world use demonstrate the effectiveness of optimizers for hyperparameter tuning [5, 12, 26]. Notable examples exist in medicine [27], finance [31] and the energy sector [6]. Hybrid optimizers have also demonstrated favorable results when applied to resolving real-world challenges [8, 21, 30].

While there has been extensive research on the application of hybrid optimizers in various domains, their combination with metaheuristic optimizers and classifiers has been very rarely seen in addressing social media issues. This offers a chance to create novel solutions that take advantage of machine learning and optimization techniques for tackling these problems.

2.1 TF-IDF Text Mining

An efficient method for handling text summary is using TF-IDF [4]. The TF-IDF is a statistical measure that quantifies a word's significance to a document by giving it a number to represent it within a collection or corpus. This can be explained using the following equations: Eq. (1) is used to determine the Term

Frequency (TF):

$$\mathrm{TF}(x, y) = \frac{Total\ count\ of\ words\ x\ in\ document\ y}{Total\ count\ of\ all\ words\ in\ document\ y} \tag{1}$$

Eq. (2) is used to calculate Inverse Document Frequency (IDF) on a corpus of text:

$$\mathrm{IDF}(x, Y) = \log \frac{Total\ count\ of\ documents\ in\ corpus\ Y}{Total\ count\ of\ all\ documents\ that\ contain\ word\ x} \tag{2}$$

Lastly, the relationship between TF and IDF is established in Eq. (3):

$$\text{TF} - \text{IDF} = \text{TF} \times \text{IDF} \tag{3}$$

The frequency of a word in the corpus offsets the TF-IDF value, which rises according to the number of times it appears in the document. This helps to mitigate the effect of certain words being more common than others. This basically means that if we have two words, for example, "the" and "horizontal," as the first one is more commonly appearing in more documents, its factor will be considerably lower. Values range from zero to one, and when all calculations are done, sentences get sorted by the sum of values from their words.

2.2 AdaBoost Classification

Boosting has proven to be an extremely effective method for resolving the two-class classification issue. One of the best methods for boosting is using AdaBoost [9], which is considered one of the best-starting classifiers. The prevalent interpretation of boosting within the statistics field considers AdaBoost as a functional gradient descent technique. In the two-class scenario, AdaBoost just needs each weak classifier's performance to be better than random chance. Through the iterative process of integrating several weak classifiers, the AdaBoost algorithm attempts to mimic the Bayes classifier by combining multiple weak classifiers.

Beginning with the unweighted training sample, AdaBoost constructs a classifier that generates class labels, such as a classification tree. A training data point's weight is enhanced (boosted) if it is incorrectly categorized. The total error is calculated using the total error from the formula in Eq. (4):

$$\text{Total error} = \frac{\sum_{i=1}^{n} \omega_i \cdot I(y_i \neq k_m(x_i))}{\sum_{i=1}^{n} \omega_i} \tag{4}$$

where n is the number of samples, k is prediction made in iteration m for i-th component. The I represents the state of prediction where 0 is considered incorrect and is considered correct prediction.

Using the new, non-equal weights, a second classifier is generated. The process is then repeated, with the weights of the incorrectly categorized training data enhanced. In most cases, this allows one to develop a large number of classifiers. Every classifier is given a score in the form of weights, and the linear combination of the classifiers from every stage is the final classifier. Weights can be calculated using the formula in Eq. (5):

$$\alpha = \frac{1}{2}\log\left(\frac{1 - \textit{Total error}}{\textit{Total error}}\right) \tag{5}$$

Weight α is linked to a weak classifier, which is utilized to ascertain the contribution. This is dependent on the total error that was found previously. Ultimately, the classifier's contribution may be ascertained with the formula presented in Eq. (6):

$$w_i = w_i \cdot e^{(\alpha \cdot I(y_i \neq k_m(x_i)))} \tag{6}$$

The final classifier will be defined as a linear combination of all previous classifiers. The final result can be seen in Eq. (7):

$$\text{Final classifier} = \arg\max_{z} \sum_{m=1}^{M} \alpha \cdot I(k_m(x_i) = z) \tag{7}$$

3 Methods

3.1 The Original VNS Algorithm

Local search methods for combinatorial optimization operate by iteratively modifying an initial solution at the local level, increasing the objective function's value until a local optimum is attained. In other words, until no further improvements can be identified, an improved solution of x in the neighborhood $N(x)$ is acquired at each repetition. Unlike most other local search techniques, VNS [25] does not adhere to a single trajectory. Rather, it investigates progressively farther reaches of the existing, dominant solution, switching to a new one only in the event that an enhancement is discovered. The majority of the variables in the incumbent solution are already at their optimal levels, which is a benefit that this technique helps maintain and may be used to identify prospective neighboring solutions. To attain local optima, these surrounding solutions are also subjected to a repetitive application of a local search procedure.

VNS works in the following way: firstly, we should assume that there is a finite set of neighborhood structures *Nk*, where k = 1, 2,... *Kmax*. VNS will go through every neighborhood structure in search of x, from the first structure to the last, iterating three steps:

- Create a random x' from the possible set of x in the given neighborhood.
  This is done in order to prevent cycling.
- Use some form of local search where x' is already given a solution, and it will obtain the local optima, for example, x''.
- If that local optimum is better than the already-given solution, that one is set as the new solution, and the search continues in the given neighborhood. If not, it will go to the next neighborhood structure.

There are also numerous other reasons that can stop this sequence of iterations, such as the maximum number of iterations or the maximum CPU-allowed time.

3.2 Modified VNS Algorithm

The original VNS demonstrates impressive outcomes when tackling optimization tasks. However, given that the VNS is a fairly old algorithm, the combination between the base VNS and techniques that could potentially boost its performance has not yet been fully explored. A potential improvement in optimization as well as computation performance can be introduced by incorporating quasireflexive learning (QRL) [29] and a generalized island model [10] approach into the baseline VNS algorithm.

The QRL method is integrated into the initialization of the population. The initial 50% of the population is generated using standard VNS techniques. The remaining agents are then generated using QRL. This helps boost diversity even prior to the first iteration of the optimization. When applying QRL each agent is synthesized using Eq. (8)

$$X_j^{qr} = \mathrm{rnd}\left(\frac{\mathrm{lb_j} + \mathrm{ub_j}}{2}, \mathrm{x_j}\right) \tag{8}$$

here, *rnd* has the role to select the random value within $\left[\frac{lb_j+ub_j}{2}, x_j\right]$ limits.

Originally proposed as a way of improving the performance of the GA, the island-based approach is used to improve the diversity of solutions. This is accomplished by using a number of sub-populations. By introducing a sub-structure into the population, parallelization mechanisms can be leveraged to distribute optimization tasks across multiple computational devices. Recombination is conducted between agents inside of an existing subpopulation. To allow communications between sub-population the modified algorithm integrates a migration procedure that combines agents from each sub-population. In the first half of iterations, the worst agent from each subpopulation is replaced by a random agent from the opposite population. In the later part of the optimization, the worst solution is replaced by the best-performing agent from the opposite sub-population.

The introduced algorithm is dubbed the multi-population VNS (MPVSN). The pseudocode for the introduced modified metaheuristic is provided in Algorithm 1.

4 Simulation Configuration

To validate the proposed methodology a real-world dataset accessible on Kaggle[1] is utilized to carry out a comparative analysis between optimizes. Data is split into training data (70%) and withheld portion of the data (30%) utilized for evaluations. Several metaheuristic optimizes have been subjected to a comparative analysis under identical conditions tasked with selecting optimal AdaBoost hyperparameters. Parameters selected for optimization and their respective ranges are determined empirically and provided in Table 1.

[1] https://www.kaggle.com/datasets/saurabhshahane/cyberbullying-dataset.

Algorithm 1 MPVNS procedure pseudo-code.

```
Produced 50% of population P using VNS initialization
Apply QRL to generate remaining agents in population P
Shuffle population and randomly assigned to two sub-populations
Define initial shaking step solution x
Set T = max_iterations
Set t = 0
while T > t do
  for Each subpopulation P_sub do
    for Agent A in P_sub do
      Evaluate agents and rank according to objective function
      Apply shaking procedure generate
      Apply local search
      Apply Neighborhood shift
    end for
  end for
  if t > T/2 then

    Replace the worse solution in each sub-population with a random agent from
    another sub-population
  else
    Replace the worst solution in each sub-population with the best agent from
    another sub-population
  end if
  t = t + 1
end while
```

Table 1. Adaboost hyperparameters and ranges.

Adaboost parameter	range
count of estimators	[10, 30]
depth	[1, 5]
learning rate	[0.01, 2]

Algorithms included in the comparisons are the introduced modified algorithm MPVNS as well as the baseline metaheuristic VNS [25]. Well-established optimizers are also included such as the GA [24], PSO [14], and WOA [23]. Recently introduced optimizers have also been evaluated the SCHA [3], RSA [1], ChOA [15]. Finally, the SSA [22] algorithm is included as it has demonstrated good performance in preceding research. Population sizes are set to 10 agents and carried out in 15 iterations. Simulations are carried out through 30 independent runs to ensure experimental viability.

Optimized models are evaluated using standard classifier metrics including accuracy 9, precision 10, specificity 11, f1-score 12. The Cohen's kappa 13 metric is used as the objective for the optimization. The error rate 14 is tracked as the indicator function.

$$Accuracy = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \tag{9}$$

$$Precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{10}$$

$$Specificity = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{11}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{12}$$

$$\text{Cohen}'\text{s kappa} = \frac{\text{Model Prediction} - \text{Expected Class}}{1 - \text{Expected Class}} \tag{13}$$

$$\text{Error rate} = 1 - \text{Accuracy} \tag{14}$$

A flowchart of the proposed experimental framework is provided in Fig. 1 to visually illustrate the optimization process.

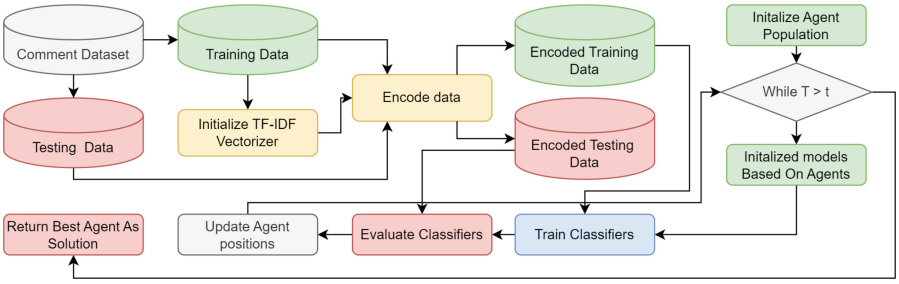


Fig. 1. Optimization framework flowchart.

5 Simulation Outcomes

A comparison between the model optimized by each metaheuristic based on objective scores is presented in Table 3. The opposed MPVNS showcases admirable abilities attaining the best outcome of 0.677586 for the best optimized model. A notable comparison between algorithm stability can also be established. The ChOA attained the highest stability rates in comparison to other optimizers. However, the ChOA high stability rates stem from an inability to overcome a local optimum in the search space resulting in lower outcomes overall in comparison to the modified algorithm (Table 2).

Table 2. Objective (Cohen's kappa) outcomes across evaluations.

Method	Best	Worst	Mean	Median	Std	Var
AB-MPVNS	**.677586**	**.663662**	**.668630**	**.667816**	.004374	1.91E-5
AB-VNS	.670854	.651357	.658858	.656391	.006578	4.33E-5
AB-GA	.665336	.644666	.657587	.659700	.005834	3.40E-5
AB-PSO	.672959	.649569	.658345	.657722	.006305	3.98E-5
AB-WOA	.663065	.644112	.653809	.654333	.006279	3.94E-5
AB-SSA	.670570	.652362	.661067	.659984	.005482	3.01E-5
AB-SCHA	.668246	.646745	.655158	.651986	.007295	5.32E-5
AB-RSA	.666797	.648249	.656949	.656763	.006142	3.77E-5
AB-ChOA	.665446	.651357	.657310	.656457	**.004132**	**1.71E-5**

Comparisons are provided in terms of indicator function metrics between models optimized by each mechanistic. The introduced MPVNS optimized attained the best outcomes in terms of indicator metrics for the best, worst, mean, and median executions, with the best score being .076942. In terms of stability for the indicator function, the SSA algorithm demonstrates the highest rates. However, the algorithm is hindered by a local optima, not managing to find the best results despite high stability.

Table 3. Indicator (error rate) outcomes across evaluations.

Method	Best	Worst	Mean	Median	Std	Var
AB-MPVNS	**.076942**	**.081148**	**.079478**	**.078921**	.001919	3.68E-6
AB-VNS	.077684	.084117	.080870	.081148	.002042	4.17E-6
AB-GA	.080158	.084612	.082540	.082261	.002117	4.48E-6
AB-PSO	.077932	.084364	.080870	.080529	.001790	3.20E-6
AB-WOA	.079169	.083622	.082323	.082632	.001927	3.71E-6
AB-SSA	.078674	.080406	.080591	.080653	**.001314**	**1.73E-6**
AB-SCHA	.080158	.083127	.081952	.082385	.001382	1.91E-6
AB-RSA	.080158	.086591	.081643	.081395	.002278	5.19E-6
AB-ChOA	.079664	.084117	.081055	.080901	.001574	2.48E-6

Further visual comparisons between optimizer stability are provided in Fig. 2 where optimizer outcome distributions for the objective and indicator functions are depicted. The introduced optimizer outperforms the base algorithm according in both cases demonstrating a higher rate of stability. While the ChOA algorithm attained the highest comparable stability in terms of objective function, and the SSA algorithm attained the highest stability in terms of indicator function these algorithms fail to attain the best results in

comparison to other algorithms suggesting that a local minimum exists within the search that combined with the optimizers lack an appropriate trade-off between exploration and exploitation limits overall performance outcomes.

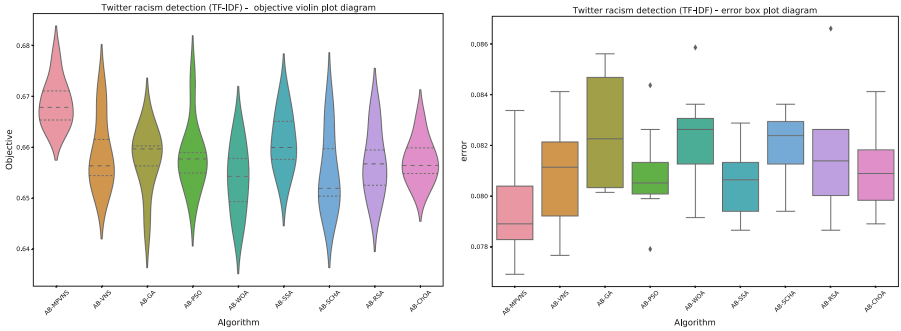


Fig. 2. Distributions in terms of objective and indicator functions.

A comparative analysis between the best-performing models optimized by each algorithm is provided in Table 4. In terms of accuracy, the introduced modified algorithm-optimized models attained the best outcomes with a score of .923058. The introduced optimized also attained the best decent weighted averages and macro averages. The VNS attained the best results in terms of non-racism detection and macro average outcomes. A clear improvement can be observed in terms of the performance of the modified algorithm over the baseline metaheuristic for this specific application.

Convergence rates provide valuable insight into to balance between exploration and exploitation of a metaheuristic optimizer. Striking a balance between mechanisms is essential, as premature convergence can lead to sub-optimal outcomes. Similarly, an algorithm that fails to focus on promising regions of the search space will fail to locate an optimum. Comparison in terms of convergence between the analyzed optimizes are provided in Fig. 3. The introduced optimize manages to locate a promising region in iteration 10 where a minimum in terms of both objective and indicator functions is located. Other optimizers struggle with local optimums failing to converge towards a more promising search space.

To support experimental repeatability, parameters selected made optimizers for are tracked and provided in Table 5 for the best performing optimizers. Further details on the best-performing model in the form of a PR curve and confusion matrix are provided in Fig. 4.

Table 4. Detailed comparisons between best-performing models constructed optimizers.

Method	metric	non-racism	racism	accuracy	macro avg	weighted avg
AB-MPVNS	precision	.946784	.765152	**0.923058**	0.855968	**0.920227**
	specificity	.964068	.683587	**.923058**	**.823828**	**.923058**
	f1-score	.955348	**.722073**	**.923058**	**.838711**	**.921240**
AB-VNS	precision	.944460	**.769981**	.922316	**.857220**	.918949
	specificity	.965807	.668359	.922316	.817083	.922316
	f1-score	.955014	.715580	.922316	.835297	.920005
AB-GA	precision	.945569	.750469	.919842	.848019	.917042
	specificity	.961460	.676819	.919842	.819140	.919842
	f1-score	.953448	.711744	.919842	.832596	.918108
AB-PSO	precision	.945961	.762357	.922068	.854159	.919116
	specificity	.963779	.678511	.922068	.821145	.922068
	f1-score	.954787	.717995	.922068	.836391	.920164
AB-WOA	precision	.942857	.767258	.920831	.855058	.917182
	specificity	.965807	.658206	.920831	.812007	.920831
	f1-score	.954194	.708561	.920831	.831378	.918279
AB-SSA	precision	.945915	.758034	.921326	.851975	.918444
	specificity	.962909	.678511	.921326	.820710	.921326
	f1-score	.954337	.716071	.921326	.835204	.919499
AB-SCHA	precision	**.947098**	.744954	.919842	.846026	.917541
	specificity	.959722	**.686971**	.919842	.823347	.919842
	f1-score	.953368	.714789	.919842	.834078	.918484
AB-RSA	precision	.946332	.747681	.919842	.847006	.917286
	specificity	.960591	.681895	.919842	.821243	.919842
	f1-score	.953408	.713274	.919842	.833341	.918297
AB-ChOA	precision	.944839	.756190	.920336	.850515	.917256
	specificity	.962909	.671743	.920336	.817326	.920336
	f1-score	.953789	.711470	.920336	.832629	.918358
	support	3451	591			

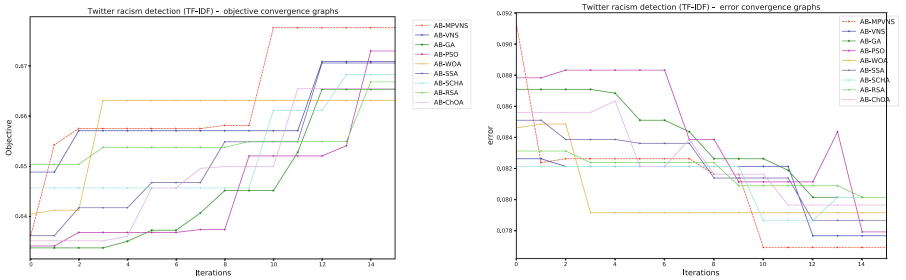


Fig. 3. Objective and indicator function distribution diagrams.

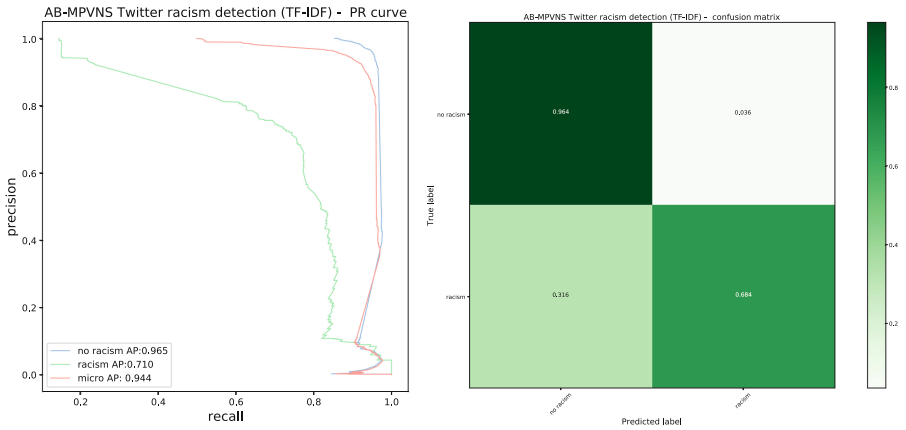


Fig. 4. Best performing model PR plot and confusion matrix.

Table 5. Parameters selected by each optimizer for optimized models.

Methods	Number of estimators	Depth	Learning rate
AB-MPVNS	25	4	.721284
AB-VNS	21	4	.740920
AB-GA	13	6	.668236
AB-PSO	22	5	.667767
AB-WOA	26	4	.746295
AB-SSA	18	5	.762458
AB-SCHA	19	4	.830139
AB-RSA	22	4	.815819
AB-ChOA	18	4	.801553

6 Conclusion

The advent of social media platforms has resulted in a significant shift in how individuals interact and communicate. However, technology has also played a significant role in the noticeable increase in the spread of hate speech incidents. Using artificial intelligence to filter content has both good and bad sides. Although it can help with maintaining safe online environments, it also introduces risks related to transparency, censorship, bias, and control over digital content. To address these issues, it is imperative to integrate a wide variety of professions to ensure that AI systems are developed and used responsibly, respecting both individual rights and societal needs. This study presents an extensive approach to classifying racist material. The framework improves the accuracy of identifying racially offensive content. The outcomes show how effective this strategy is in recognizing and categorizing racist comments on Twitter, enabling automatic content moderation, and creating a more inclusive online community. Favorable results are further guaranteed by the addition of a modified metastasis optimization to classifier optimization. The best-performing classifiers in real-world data analysis reach an accuracy of .92305. Several challenges exist within this study. Twitter policy changes make new data less accessible for independent researchers, making new data collection difficult. Furthermore, processing loads of optimization constrain the amount of optimizers that could be evaluated as well as the population sizes and iterations for each optimization.

Future works hope to refine the methodology further and expand comparative analysis as computation moves forward. Additionally, different encoding schemes hope to be explored, and applications for the introduced optimizer are found in other pressing areas of research.

References

1. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W., Gandomi, A.H.: Reptile search algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert Syst. Appl. **191**, 116158 (2022)
2. Adam, S.P., Alexandropoulos, SA.N., Pardalos, P.M., Vrahatis, M.N.: No free lunch theorem: a review. In: Demetriou, I., Pardalos, P. (eds.) Approximation and Optimization. SOIA, vol. 145, pp. 57–82. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-12767-1_5
3. Bai, J., et al.: A sinh cosh optimizer. Knowl.-Based Syst. **282**, 111081 (2023)
4. Christian, H., Agus, M.P., Suhartono, D.: Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF). ComTech Comput. Math. Eng. Appl. **7**(4), 285–294 (2016)
5. Cuk, A., et al.: Tuning attention based long-short term memory neural networks for Parkinson's disease detection using modified metaheuristics. Sci. Rep. **14**(1), 4309 (2024)
6. Damaševičius, R., et al.: Decomposition aided attention-based recurrent neural networks for multistep ahead time-series forecasting of renewable power generation. PeerJ Comput. Sci. **10**, e1795 (2024)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)

8. Gajevic, M., Milutinovic, N., Krstovic, J., Jovanovic, L., Marjanovic, M., Stoean, C.: Artificial neural network tuning by improved sine cosine algorithm for healthcare 4.0. In: Proceedings of the 1st International Conference on Innovation in Information Technology and Business (ICIITB 2022), vol. 104, p. 289. Springer Nature (2023)
9. Hastie, T., Rosset, S., Zhu, J., Zou, H.: Multi-class adaboost. Stat. Interface **2**(3), 349–360 (2009)
10. Izzo, D., Ruciński, M., Biscani, F.: The generalized island model. In: Parallel Architectures and Bioinspired Algorithms, pp. 151–169. Springer (2012)
11. Jaiswal, G., Rani, R., Mangotra, H., Sharma, A.: Integration of hyperspectral imaging and autoencoders: benefits, applications, hyperparameter tunning and challenges. Comput. Sci. Rev. **50**, 100584 (2023)
12. Jovanovic, L., et al.: Tuning XGBoost by planet optimization algorithm: an application for diabetes classification. In: Bindhu, V., Tavares, J.M.R.S., Vuppalapati, C. (eds.) Proceedings of Fourth International Conference on Communication, Computing and Electronics Systems. LNEE, vol. 977, pp. 787–803. Springer, Singapore (2023). https://doi.org/10.1007/978-981-19-7753-4_60
13. Kaur, A., Chopra, D.: Comparison of text mining tools. In: 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), pp. 186–192. IEEE (2016)
14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN 1995 - International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
15. Khishe, M., Mosavi, M.R.: Chimp optimization algorithm. Expert Syst. Appl. **149**, 113338 (2020)
16. Kumar, A., Sharma, R., Bedi, P.: Towards optimal NLP solutions: analyzing GPT and LLaMA-2 models across model scale, dataset size, and task diversity. Eng. Technol. Appl. Sci. Res. **14**(3), 14219–14224 (2024)
17. Liu, X., et al.: GPT understands, too. AI Open (2023)
18. Lozano, E., Cedeño, J., Castillo, G., Layedra, F., Lasso, H., Vaca, C.: Requiem for online harassers: identifying racism from political tweets. In: 2017 Fourth International Conference on eDemocracy & eGovernment (ICEDEG), pp. 154–160. IEEE (2017)
19. Matamoros-Fernández, A., Farkas, J.: Racism, hate speech, and social media: a systematic review and critique. Telev. New Media **22**(2), 205–224 (2021)
20. Mathew, T.V.: Genetic algorithm. Report Submitted at IIT Bombay **53**, 18–19 (2012)
21. Minic, A., et al.: Applying recurrent neural networks for anomaly detection in electrocardiogram sensor data. Sensors **23**(24), 9878 (2023)
22. Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., Mirjalili, S.M.: Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. **114**, 163–191 (2017)
23. Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. **95**, 51–67 (2016)
24. Mirjalili, S.: Genetic algorithm. In: Evolutionary Algorithms and Neural Networks. Studies in Computational Intelligence, vol. 780, pp. 43–55. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-93025-1_4
25. Mladenović, N., Hansen, P.: Variable neighborhood search. Comput. Oper. Res. **24**(11), 1097–1100 (1997)
26. Petrovic, A., Antonijevic, M., Strumberger, I., Jovanovic, L., Savanovic, N., Janicijevic, S.: The XGBoost approach tuned by TLB metaheuristics for fraud detection. In: Proceedings of the 1st International Conference on Innovation in Information Technology and Business (ICIITB 2022), vol. 104, p. 219. Springer Nature (2023)
27. Pilcevic, D., et al.: Performance evaluation of metaheuristics-tuned recurrent neural networks for electroencephalography anomaly detection. Front. Physiol. **14**, 1267011 (2023)

28. Purnomo, H., Gonsalves, T., Mailoa, E., Santoso, F.J., Pribadi, M.R., et al.: Meta-heuristics approach for hyperparameter tuning of convolutional neural network. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi) **8**(3), 340–345 (2024)
29. Rahnamayan, S., Tizhoosh, H.R., Salama, M.M.: Quasi-oppositional differential evolution. In: 2007 IEEE Congress on Evolutionary Computation, pp. 2229–2236. IEEE (2007)
30. Salb, M., et al.: Enhancing internet of things network security using hybrid CNN and XGBoost model tuned via modified reptile search algorithm. Appl. Sci. **13**(23), 12687 (2023)
31. Stankovic, M., Bacanin, N., Zivkovic, M., Jovanovic, L., Mani, J., Antonijevic, M.: Forecasting ethereum price by tuned long short-term memory model. In: 2022 30th Telecommunications Forum (TELFOR), pp. 1–4. IEEE (2022)
32. Yang, L., Shami, A.: On hyperparameter optimization of machine learning algorithms: theory and practice. Neurocomputing **415**, 295–316 (2020)

Lower Limb Prosthetics Activity Recognition Using Optimized CNN-LSTM Model

M. Jeyasudha[1](✉), S. Prakash[1], and G. Prakash[2]

[1] Department of Electrical and Electronics Engineering, Bharath Institute of Higher Education and Research, Chennai 600073, Tamil Nadu, India
{jeyasudha.cbcs,prakash.eee}@bharathuniv.ac.in
[2] Department of Electrical and Electronics Engineering, Sri Sairam Institute of Technology, West Tambaram, Chennai, Tamilnadu, India
prakash.eee@sairamit.edu.in

Abstract. Lower Limb Prosthetics Activity Recognition entails identifying and categorizing human movements related to activities using the lower limbs. Still, the detection of lower limb prosthesis activity is hindered by intricate movements, inconsistent user patterns, and challenges in real-time processing. To overcome hurdles, one can use advanced sensors, personalized training, and ensemble learning approaches to improve accuracy. The present study addresses the issue by employing the Optimized CNN-LSTM model, which enhances the accuracy of activity identification in lower limb prostheses. The suggested approach comprises four main steps: pre-processing, feature extraction, feature selection, and classification. The gathered pictures undergo pre-processing using an enhanced Gaussian Bilateral Filter and Complete Ensemble Empirical Mode Decomposition (CEEMD). Features are derived from pre-processed data using a Dual-tree Complex Wavelet Transform (DTCWT). The extracted features are then transferred to the Feature categorization procedure. The classification step is performed using the Optimized CNN-LSTM technique. This model has been improved using the Adapted Bat Algorithm to enhance its performance. After the classification, the feedback level is included in the proposed approach. The suggested model is implemented using the PYTHON programming language, and its performance is assessed using metrics such as accuracy, precision, recall, F-score, specificity, sensitivity, MCC, NPV, FPR, and FNR. Suggested optimized CNN-LSTM achieves higher accuracy (99.23%) and precision (98.83%) compared to current techniques.

Keywords: Lower Limb Prosthetics · Activity Recognition · Dual-tree Complex Wavelet Transform · Adapted Bat Algorithm · Optimized CNN-LSTM

1 Introduction

Human locomotion is a very intricate form of movement that may be understood by studying the kinematics and muscle activity of the limbs during their different motions. Human gait analysis is a process that discovers and examines issues related to walking and posture, such as abnormal loads and muscular weakness [1]. These issues cannot be

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 232–255, 2025.
https://doi.org/10.1007/978-3-031-86296-0_17

measured by regular clinical examinations. Gait may be described as a certain pattern of movement that occurs within a narrow range of speeds. It is used as a valuable diagnostic tool in several domains, including healthcare, sports, daily activities, and prostheses. Consequently, gait can have a substantial influence on an individual's Quality of Life (QOL) [2]. Activity recognition in lower limb prosthetics is the utilization of sensors and algorithms to detect and analyze the motions and activities executed by an individual utilizing a prosthetic limb [3].

The system can differentiate between different actions, such as walking, running, or standing still, by examining data collected from accelerometers, gyroscopes, and other sensors included in the prosthesis. This technique aims to enhance the functionality of lower limb prostheses by providing users with instant feedback and enabling a more genuine and flexible prosthetic experience, eventually enhancing the ability to move and the overall quality of life for those who have lost a leg [4]. An individual's prosthesis should be tailored and constructed based on their own aesthetic and practical requirements. There are three main groups into which upper limb prostheses fall: Externally Powered (myoelectric) devices, Passive devices, and Body Powered devices [5].

Lower-extremity prostheses offer replacements for different degrees of amputation. The types of amputations mentioned are toe, transtibial prosthesis, knee disarticulation, Syme's amputation, hip disarticulation, foot, transfemoral prosthesis, and partial foot [6]. Recently, academics have shown interest in Human Activity Recognition (HAR) due to developments in artificial intelligence techniques, computer vision, the Internet of Things and presence of wearable sensors [7]. The Human Activity Recognition (HAR) system is capable of identifying a diverse range of human movements, including jogging, sitting, standing, walking, sleeping, bathing, cooking, and driving [8].

A wide range of HAR applications may be observed in several fields, applications for tele-immersion, smart homes with improved security, and healthcare monitoring. The main obstacle with powered lower limb prostheses stems from the restricted level of direct manipulation that amputees possess over these devices [9]. To challenge this issue, many control mechanisms are being investigated, such as surface electromyography (sEMG), mechanical sensors, or a fusion-based approach. The electromyographic signal (EMG), produced by muscle action, is a crucial physiologic signal for brain regulation [10]. Surface electromyography (EMG) electrodes capture muscle activity data, enabling amputees to effortlessly operate both passive and powered prostheses. The use of surface electromyography (sEMG) improves the sensitivity of lower limb prostheses, allowing users to control and engage with their artificial limbs more instinctively and efficiently [11].

The Lower Limb Prosthetics Activity Recognition system utilizes sophisticated technologies such as Deep Learning (DL) Models and Machine Learning to improve its functioning. These models make use of data collected by built-in sensors like accelerometers and gyroscopes [12]. DL Models, often convolutional neural networks (CNNs), autonomously extract complex information from sensor input, optimizing for both efficiency and accuracy in identifying different behaviors such as walking or running [13]. Machine Learning (ML) methods, such as support vector machines or decision trees, use past data to identify trends and make accurate predictions for activity detection. The integration of these models enables a comprehensive and adaptable system, which

continuously improves the control of prosthetics by analyzing the user's intentions in real time [14]. Employing surface electromyography (sEMG) data enhances the precision of activity detection. Through examination of muscle activity data, prosthetic system acquires knowledge about the user's motor goals, hence enhancing the lower limb prosthetic experience by making it more intuitive and responsive [15].

The main contributions of the paper are as follows:

- To improve performance of the Optimized CNN-LSTM model by integrating the Adapted Bat Algorithm. This optimization approach enhances the model's parameters, allowing it to dynamically adapt to the complexities of lower limb prosthesis activities and enhance overall accuracy.
- Advanced signal processing methods such as an upgraded Gaussian Bilateral Filter, CEEMD, and DTCWT are used in the study. These methods improve preprocessing and feature extraction, making the model more capable of handling various and complicated data sources.
- By effectively combining CNN and LSTM layers, the paper uses their complimentary characteristics for robust lower limb prosthesis activity identification.
- Finally, feedback of the abnormal activity is displayed to the patient.

The remaining sections are organized as follows: In Sect. 2, the relevant studies and literature reviews are discussed. Proposed framework is discussed in Sect. 3. Section 4 explores recorded outcomes by Result and Discussion, whereas Sect. 5 provides a concluding analysis of this investigation.

2 Literature Review

Some of the recent research works related to the Lower Limb Prosthetics Activity Recognition using Optimized CNN-LSTM Model were reviewed in this section

Vijayvargiya *et al.* (2022) [16] provided hybrid deep-learning models for lower limb activity detection. Electromyography (EMG) sensors detect the signal beforehand and detect movement faster, making them appropriate for automated lower limb exoskeleton or prosthetic control. After discrete wavelet transform noise suppression, overlapping windowing splits the signal. Sequence learning employs a gated recurrent unit, whereas temporal learning uses a convolutional neural network.

Slemenšek *et al.* (2023) [1] suggested wearable gait motion data collection system may categorize data into desired activities or highlight frequent hazard concerns, enhancing the subject's quality of life. Inertial forces were applied to lower limb accelerometers and gyroscopes while walking to record gait. Strain gauges measured leg muscle activity. We aimed to recognize gait activities in each gait recording using ML. When evaluated on five participants' datasets, the attention-based convolutional and recurrent neural networks method performed best.

Iqbal *et al.* (2021) [17] assumed that parallel discriminant qualities may be constructed from massive neuromechanical data from several individuals utilizing parallel and distributed computing systems. This research uses SVM, decision trees, and random forest classifiers on massive data sets. Sequential and distributed contexts rigorously

evaluate model correctness, efficiency, scalability, and speedup. Testing showed parallel processing was 3.9x quicker than sequential computing without affecting accuracy.

Peng *et al.* (2020) [18] presented multilayer classifier fusion locomotion prediction for complex activities. First, steady and transitory motion was defined. Gait backtracking was used for steady-state identification and motion prediction. Merging sEMG and inertial sensors improved steady-state recognition. With almost 97.8% accuracy, the random forest classifier identified five frequent locomotor modes. Only sEMG signal was recorded during transitory condition, the first step of movement, due to sliding window length. Finally, a rudimentary HMM model predicted movement using previous knowledge and observations.

Zhang and Tao (2022) [19] examined lower limb motion and collected sEMG signals of five gaits: going upstairs without weight, downstairs, upstairs with weight, and walking on a flat surface without weight. Using sEMG signal characteristics, a CNN is built and a new lower limb motion pattern recognition method is demonstrated. The recommended method is compared to several classification and identification methods for accuracy and work features. The experimental results imply that CNN using the feature set as input improves prediction model representation and pattern recognition accuracy.

Wang (2021) [20] explained CNN deep learning research on lower limb prosthesis motion recognition intention. The motion pattern was modeled using CNN. Before changing the affected side's movement mode, the sensor was linked to the healthy side. Classifiers retrieved and categorized variables to appropriately characterize impaired movement intention. Deep learning techniques that automatically recognize and extract information can enhance intelligent lower limb prosthesis control and enable seamless motion mode changeover.

Vijayvargiya *et al.* (2021) [9] investigated kinesiology, diagnose neuromuscular illnesses, and talk to machines. Loud signals are confusing. WD-EEMD separates lower limb activity from sEMG data in healthy and knee-dysfunctional people to solve this problem. WGN and muscle signals are reduced via wavelet de-noising. After filtering PLI and BW sounds, an Ensemble Empirical Mode Decomposition yields nine time-domain characteristics. Final LDA classifier performance parameters are determined by 3-fold cross-validation. Compare pre-processing methods like WD-EEMD.

Hussain *et al.* (2020) [3] categorized feature space into amputee-specific activities using sEMG signal characteristics. This work used bispectrum magnitude, a novel feature extraction approach that is invariant to change, to improve activity detection performance. Eleven lower limb sensors recorded sEMG signals during walking and ramp workouts in six subjects: one unilateral transtibial, four able-bodies, and one unilateral transfemoral amputee. Support vector machine and linear discriminant analysis correctly categorized multi-class sEMG signals.

2.1 Problem Statement

See the Table 1.

Table 1. Research gaps from the existing works

Author name	Aim	Technique Used	Pros	Cons
Vijayvargiya *et al.* [16]	Hybrid DL Models for Lower Limb Activity Recognition	**CNN, LSTM, GRU, DWT**	Improved lower limb prostheses control accuracy and responsiveness	Complex model implementation
Slemenšek *et al.* [1]	System for Wearable Gait Data Acquisition	**CNN, RNN**	Improved quality of life with gait activity categorization	Required robustness testing
Iqbal *et al.* [17]	Learning Parallel Discriminant Features for Lower Limb Prosthesis Intent Recognition	**SVM, DT, RF**	Recognizing lower limb movement modalities without interference using parallel computing	Provide a formidable challenge for conventional technologies
Peng *et al.* [18]	Lower Limb Prosthesis Locomotion Prediction	**RF, Light GBM, HMM**	Anticipating movement improves gait control in complicated contexts	Transient-state recording limitation
Zhang and Tao [19]	Propose sEMG-based lower limb motion pattern recognition	**CNN**	Improved recognition accuracy over older approaches	Low data, and low recognition accuracy in modern technologies
Wang [20]	CNN deep learning for lower limb prosthesis motion recognition intention	**CNN**	Identify lower limb amputation patients' mobility intentions at high rates	Dependence on sensor location, terrain-specific

(*continued*)

Table 1. (*continued*)

Author name	Aim	Technique Used	Pros	Cons
Vijayvargiya *et al.* [9]	Classify lower limb activity from WD-EEMD pre-processed sEMG data	**WD-EEMD**	Solve sEMG signal processing issues, notably noise	Sophisticated preprocessing technique
Hussain *et al.* [3]	Discriminant characteristics for amputee activity identification using sEMG data	**Novel feature extraction using bispectrum magnitude**	Address sEMG signal non-Gaussianity and amputee mobility	Small sample size

3 Proposed Methodology

Activity detection systems and cutting-edge prosthetic technologies are combined in the developing field of lower limb prosthetics. This research focuses on utilizing a deep learning model to recognize lower limb prosthesis activity. The model is built through many steps as indicated below: (i) Data Acquisition (ii) Pre-processing approach (iii) Feature Extraction procedure (iv) Feature selection process (v) Feature classifier mechanism. Figure 1 shows the overall architecture of the proposed model.

3.1 Data Acquisition

The unprocessed data is gathered in the picture format. Image acquisition, in the setting of image processing, refers to the process of acquiring an image from an external source to perform subsequent processing on it. The acquisition of a picture is always the initial stage in the workflow, as no further processes can be carried out without it. The work employs openly accessible statistics from UCI to categorize lower limb activity. Data acquisition involves acquiring information from multiple sources. In lower limb prosthetics activity recognition and related topics, data collection entails collecting human movement data using sensors or other measurement equipment. Accelerations, angles, electromyographic signals, and other lower limb physiological characteristics can be included. Data acquisition collects complete and reliable data for analysis, modeling, and interpretation to discover trends, make educated decisions, or build applications like prosthetic device activity identification systems. Sensors or measuring devices collect lower limb data like accelerations, angles, and electromyographic impulses. The goal is to get complete lower-limb movement data. The data is critical for analyzing, modeling, and comprehending patterns, especially in lower limb prosthetics activity recognition, where accurate information improves functioning and adapts prosthetic devices to users' activities.

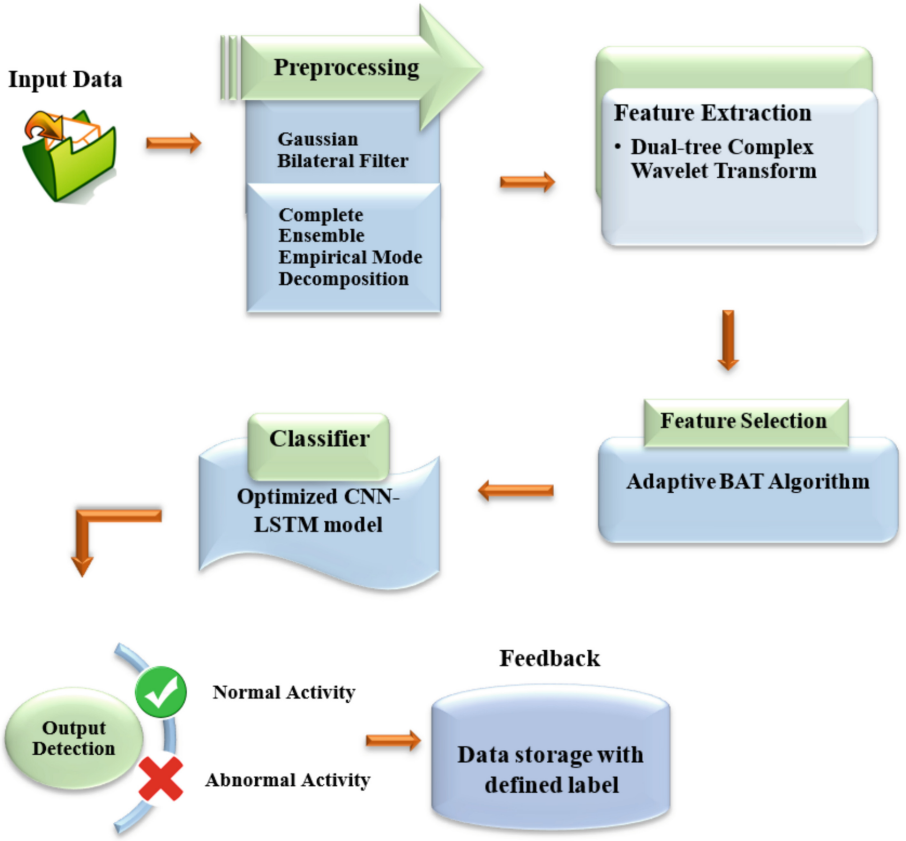


Fig. 1. Overall Architecture of the proposed model

3.2 Pre-Processing

The acquired data image is pre-processed using an Adaptive Bilateral Filter and Complete Ensemble Empirical Mode Decomposition. Pre-processing in lower limb activity identification encompasses the improvement of raw data just before analysis. Efficient pre-processing improves the precision of deep learning models in identifying and categorizing lower limb activities using sensor data.

3.2.1 Gaussian Bilateral Filter

Bilateral filters are a kind of non-linear picture-smoothing filters that minimize noise while maintaining edges. Every pixel's intensity is replaced with a computed average of the intensity values of nearby pixels. One possible source of the weight may be a Gaussian distribution. Notably, the weights depend on the radiometric changes (such color intensity fluctuations and depth distance) as well as the Euclidean distance of the pixels. The Gaussian bilateral filter is a modified version of the Gaussian filter that takes into account both the spatial closeness and intensity similarity of pixels in smoothing a

picture. It excels at maintaining the sharpness of edges while minimizing the presence of unwanted noise. The Gaussian bilateral filter and the filtered output $Op(x, y)$ at position (x, y) is given by the Eq. (1)

$$Op(x,y) = \frac{1}{\mathrm{f}} \sum_i \sum_j Inp(i,j) \cdot s(i,j) \cdot f_s(\|(i,j) - (x,y)\|) \cdot f_r(|Inp(i,j) - Inp(x,y)|) \quad (1)$$

Let Inp be the input image, $Inp(x, y)$ be the intensity of the pixel at position (x, y), and $s(x, y)$ be the spatial weight for the pixel at position (x, y).

where, f denotes the normalization factor, f_s represents the spatial Gaussian function which signifies the spatial closeness, f_r represents the range Gaussian function which signifies the intensity similarity, σ_s represents the standard deviation for the spatial proximity, and σ_τ represents the standard deviation for the intensity similarity.

The weights $w(i, j)$ are determined by the spatial distance and are often defined as to $(i, j) = e^{-\frac{d^2}{2\sigma_d^2}}$, where σ_d is a standard deviation parameter that regulates the spatial weighting. The bilateral filter takes into account both the spatial distance and the difference in intensity between pixels. The algorithm utilizes spatial and range Gaussian functions to calculate the weights assigned to the adjacent pixels. The resulting output is obtained by taking a weighted sum of the input pixel values. The parameters σ_s, σ_r, and σ_d govern the filter's behavior and must be carefully adjusted according to the image's properties and the intended filtering outcome.

3.2.2 Complete Ensemble Empirical Mode Decomposition (CEEMD)

It is a technique designed to evaluate nonlinear signals and non-stationary, such as EEG, by using data-driven and adaptive methods. However, EMD is significantly restricted by a phenomenon known as mode mixing. The phenomenon of oscillations with various amplitudes occurring in a single EMD mode, or happening of alike oscillations in multiple EMD modes, is mentioned to as mode mixing. The Complete Ensemble Empirical Mode Decomposition (CEEMD) algorithm applies EMD to a collection of signals, which includes both the original signal and additional Gaussian white noise. To address the issue of mode mixing, the use of white Gaussian noise is used. This noise is introduced to fill whole time-frequency space, allowing for exploitation of dyadic filter bank behavior of EMD. Nevertheless, CEEMD effectively addresses the problem of mode mixing, although it does so by introducing additional complications. The reconstructed signal obtained from the EEMD contains residual noise. Moreover, in the CEEMD technique, many iterations of the identical signal combined with noise might provide a varying quantity of modes. This ensemble strategy incorporates randomization, which significantly reduces the occurrence of mode mixing and enhances the stability of the decomposition process. CEEMD is a superior method for analyzing complex signal structures in many applications due to its enhanced flexibility and dependability. The mathematical representation is expressed in Eq. (2)

$$x_i(t) = x(t) + n_i(t) \quad (2)$$

Determine the average value of each IMF in the ensemble is expressed in Eq. (3)

$$IMF_j(t) = \frac{1}{n}\sum_{i=1}^{n} D_{ij}(t) \tag{3}$$

where, i denotes the indexes of the ensemble, $x(t)$ represents the signal, n is the noise signal, and j indexes the IMF.

3.3 Feature Extraction

The pre-processed data is moved on to the feature extraction procedure. Feature extraction is the procedure of converting unprocessed data into numerical characteristics that may be manipulated while retaining the information from the initial data set. It produces superior outcomes compared to directly applying ML to unprocessed data.

3.3.1 Dual-Tree Complex Wavelet Transform (DTCWT)

It is a complex signal processing approach for multiscale and multi-direction data analysis. Transform domain methods affect image processing applications. Several transformations convert the spatial image to frequency coefficients. The discrete wavelet transform (DWT) performs best. The DWT transform lacks shift-invariance, hence tiny input data changes cause substantial variations. This shifts energy distribution across DWT coefficients, reducing diagonal feature directional selectivity. Thus, Kingsbury's Dual-Tree Complex Wavelet Transform solves DWT's problems. This system has shift invariance, robust directional selectivity, perfect reconstruction, controlled redundancy, and low computing complexity. DWT and Complex wavelet transforms produce it. This method uses the DTCWT on two real Discrete Wavelet Transforms to produce complicated coefficients. Each DWT generates real and imaginary transform sections with distinct filter settings. Two real signals are obtained by inverting the real and imaginary components of two real DWTs. To recreate the final signal, the average of these two signals is used.

Using DT-CWT and a complex scaling function $\Phi(x, y)$, the input picture $f(x, y)$ is broken down into sequences of translations and dilations. This produces complex wavelet functions, or Ψ^{Θ}, which are expressed in Eq. (4)

$$f(x, y) = \sum_{l \in N^2} S_{j0,l}\Phi_{j0,l}(x, y) + \sum_{\theta \in \varnothing} \sum_{j \geq j_0} \sum_{l \in N^2} w_{j,l}^{\theta}\Psi_{j,l}^{\theta}(x, y) \tag{4}$$

In Eq. (1), $\theta \epsilon \{\pm 15^o, \pm 45^o, \pm 75^o\}$ indicates the directionality of the complex wavelet function, N stands for natural numbers, and j *and dil* for shifts and dilations, The complex wavelet coefficient is represented by wj, *dil* with $\varphi_{j0,l}(x) = \varphi_{j0,l}^{r}(x) + \sqrt{-1q}$ $\sqrt{-1\psi_{j,l}^{i}}(x)$ where the imaginary is represented by i *and* r.

At each decomposition level, applying DTCWT to an image $f(x, y)$ yields complex-valued subbands of two numbers at lower frequencies and complex-valued subbands of six numbers at higher frequencies. At angles of $\pm 15^o$, $\pm 45^o$ *and* $\pm 75^o$, the θ direction of each of these high-frequency subbands is distinct.

To enhance comprehension, the pair of complex-valued low-frequency coefficients are provided as in Eq. (5)

$$x(\lambda, Low1, u, v) = r(x(\lambda, Low1, u, v)) + ji(x(\lambda, Low1, u, v))$$
$$x(\lambda, Low2, u, v) = r(x(\lambda, Low2, u, v)) + ji(x(\lambda, Low2, u, v)) \quad (5)$$

and the six high-frequency coefficients with complex values are represented in Eq. (6)

$$x(\wedge, \theta, u, v) = r(x(\wedge, \theta, u, v)) + ji(x(\wedge, \theta, u, v)) \quad (6)$$

where,

*Low*1 *and* *Low*2 identifies sub-bands at low frequencies, $\wedge$ indicates the degree of disintegration, Θ indicates the sub-band direction, and its values are $\{-75, -45, -15, +15, +45, +75\}$. u, v shows where the coefficient is located in each sub-band, and its values change as $0 \leq u \leq \frac{N}{2\lambda} - 1$ and $0 \leq v \leq \frac{N}{2\lambda} - 1$ (Table 2).

Table 2. Mathematical expression of Mean, Median, Standard Deviation, Variance, Root Mean Square (RMS), Skewness, and kurtosis.

Metrics	Description	Mathematical expression
Mean	The mathematical average of a group consisting of two or more numbers	$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$
Standard Deviation	N be the total number of values, I_i is the individual value from 1 to N	$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(I_i - Mean)^2}{N}}$
Median	n is the number of items. If the number of items is odd, then $\left(\frac{n+1}{2}\right)^2$. If the number of items is even, then $\frac{\left(\frac{n}{2}\right)^{th} term + \left(\frac{n}{2}+1\right)^{th} term}{2}$	$\begin{cases} \left(\frac{n+1}{2}\right)^2, \text{if } n \text{ is odd} \\ \frac{\left(\frac{n}{2}\right)^{th} term + \left(\frac{n}{2}+1\right)^{th} term}{2}, \text{if } n \text{ is even} \end{cases}$
Root Mean Square (RMS)	The average magnitude of a group of data is measured, which is computed by squaring, averaging, and calculating the square root	$\sqrt{\frac{1}{n}\sum_i x_i^2}$
Variance	This statistical metric measures dataset dispersion by averaging the squared variances of each data point from the mean. It gives a numerical estimate of data point variability	$\frac{\sum (x_i - \bar{x})^2}{n-1}$

(*continued*)

Table 2. (*continued*)

Metrics	Description	Mathematical expression
Skewness	It quantifies the degree of deviation from symmetry in a dataset, indicating whether it is skewed or not	$Skew = \frac{\sum_i^N (X_i - \overline{X})^3}{(N-1)*\sigma^3}$
Kurtosis	It is a statistical measure that determines the presence of extreme values in the tails of a distribution	$Kurt = \frac{1}{N} \sum_{i=1}^{N} \left[\frac{(u_k - \overline{u})}{\sigma} \right]^4$

3.4 Feature Selection Using Adaptive Bat Algorithm

Feature selection in machine learning entails the process of selecting pertinent features to improve model performance and interpretability by removing duplicate ones. The Adaptive Bat Algorithm is employed for its ability to dynamically alter parameters, which optimizes the search process and enhances efficiency in discovering useful characteristics. Consequently, this improves the accuracy of the model in many applications.

3.4.1 Adaptive Bat Algorithm

3.4.1.1 Motivation

The basic bat algorithm (BA) guides each bat based on the best position determined so far. The algorithm's efficacy depends on the update rule. Exclusive usage of the best individual for search direction may lead to early convergence issues. Individuals are drawn to local optimums, resulting in no matching actions to overcome the limitation. Ineffective mutation mechanisms cause fast population diversity loss, leading to premature algorithm convergence. Simultaneously, the population is exploited straightforwardly, preventing the efficient use of concealed information and prior search processes. To address these issues, present an enhanced search equation and parameter adaption technique that utilizes information from best persons, nearby individuals, and past search processes.

3.4.1.2 Principe of BA

The Bat Algorithm's (BA) operation is based on the echolocation behavior of bats, which locate their prey by using SONAR or echolocation. Bats utilize the time delays and variations in loudness of received signals to sense their environment. Bats have a remarkable ability to manipulate items and locate prey by analyzing changes in their speed and location. They may also alter the sounds they produce in terms of wavelength and pulse rate. This method may move between the exploration and exploitation phases while finding the optimal location, and it has a higher early phase convergence rate. Variations in the frequency and intensity of pulse emissions during the global solution search cause the automatic switching to occur. Multi-modal optimization techniques can be solved using this algorithm.

The BA's update equation is expressed numerically as in Eq. (7)

$$BA_x^{g+1} = BA_x^g + k_x^g \tag{7}$$

where,

BA_x^{g+1} refers the x^{th} bat position in $(g+1)^{th}$ iteration, BA_x^g represents the x^{th} bat position in g^{th} iteration and k_x^g represents the x^{th} bat velocity in g^{th} iteration. Velocity of the bat in g^{th} iteration in expressed in Eq. (8)

$$k_x^g = k_x^{g+1} - \left(BA_x^g - BA_{bat}^*\right)\lambda_x \tag{8}$$

The location of the bat in the $(g+1)^{th}$ iteration is represented in Eq. (9)

$$BA_{x,bat}^{g+1} = BA_x^g(1-\lambda_x) + k_x^g + BA_{bat}^* \times \lambda_x \tag{9}$$

The variable BA_{bat}^* represents the optimal location of the bat, λ_x represents the frequency of the x^{th} bat, and k_x^{s+1} represents the velocity of the bat in the $(g+1)^{th}$ iteration.

3.4.1.3 The Exponential Weighted Moving Average (EWMA) Principle

This method calculates the average of the data with decreasing weight over time. The EWMA procedure is a monitoring technique that guarantees the management of data volatility and offers the most recent estimate of the variance rate. Equation for Exponentially Weighted Moving Average (EWMA) is given in Eq. (10)

$$BA_{EWMA}^g = \eta \times BA_x^g + (1-\eta) \times BA_{EWMA}^{g-1} \tag{10}$$

$$BA_x^g = \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \tag{11}$$

where g is a constant between 0 and 1, BA_{EWMA}^g denotes the record of the previous iteration, and BA_{EWMA}^{g-1} represents the record of the current iteration. The current forecast relies on past data that are multiplied by a weight. Adjusting the sample period allows for faster generation of knowledge about the process, hence preventing the ongoing creation of unsatisfactory outcomes.

3.4.1.4 BAT Algorithm and EWMA Principles

An adaptive algorithm is a computing process that modifies its behavior in real time by making use of predefined reward structures and existing information. The adaptive idea is used with the E-Bat algorithm, which combines EWMA and BA, to generate the update rule of the adaptive E-Bat algorithm. The E-Bat algorithm's convergence is improved by relocating the bat in accordance with its position from the previous iteration and its best position to date. This update is produced by replacing Eq. (12) with Eq. (10).

$$BA_{x,bat}^{g+1} = \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \times k_x^{g+1} + BA_{bat}^* \times \lambda_x \tag{12}$$

From the equation for velocity as represented in Eq. (8)

$$k_x^{g+1} - k_x^g = \left(BA_x^g - BA_{bat}^*\right)\lambda_x \tag{13}$$

$$\left(BA_x^g - BA_{bat}^*\right)\lambda_x = \left(\frac{3}{2}\right)^{\beta} k_x^{g+1} - \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta}{3}\right)^{\beta} k_x^g + \left(\frac{3}{2}\right)^{\beta}\left(\frac{\beta(8\beta-5)}{9}\right)k_x^{g-1} + \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta(\beta-1)(7-8\beta)}{81}\right)k_x^{s-2} \tag{14}$$

To include the adaptive idea in Eq. (12), subtract the word Bs from both sides of the equation.

$$BA_{x,bat}^{g+1} - BA_x^g\lambda_x = \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \times (1-\lambda_x) + k_x^{g+1} - \lambda_x + BA_{bat}^* \times \lambda_x \tag{15}$$

$$BA_{x,bat}^{g+1} - BA_x^g\lambda_x = \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \times (1-\lambda_x) + k_x^{g+1} - \left(BA_x^g - BA_{bat}^*\right)\lambda_x \tag{16}$$

Substitute Eq. (14) in Eq. (17)

$$BA_{x,bat}^{g+1} - BA_x^g\lambda_x = \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \times (1-\lambda_x) + k_x^{g+1} - \left\{\left(\frac{3}{2}\right)^{\beta} k_x^{g+1} - \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta}{3}\right)^{\beta} k_x^g + \left(\frac{3}{2}\right)^{\beta}\left(\frac{\beta(8\beta-5)}{9}\right)k_x^{g-1} + \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta(\beta-1)(7-8\beta)}{81}\right)k_x^{s-2}\right\} \tag{17}$$

$$BA_{x,bat}^{g+1} = BA_x^g\lambda_x + \frac{1}{\eta}\left[BA_{EWMA}^g - (1-\eta) \times BA_{EWMA}^{g-1}\right] \times (1-\lambda_x) + k_x^{g+1} - \left\{\left(\frac{3}{2}\right)^{\beta} k_x^{g+1} - \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta}{3}\right)^{\beta} k_x^g + \left(\frac{3}{2}\right)^{\beta}\left(\frac{\beta(8\beta-5)}{9}\right)k_x^{g-1} + \left(\frac{3}{2}\right)^{\beta}\left(\frac{4\beta(\beta-1)(7-8\beta)}{81}\right)k_x^{s-2}\right\} \tag{18}$$

The solutions are updated using Eq. (18) which incorporates the integration of adaptive concept, BA, and EWMA ideas.

Pseudocode for Adaptive BAT Algorithm

Algorithm: Pseudocode for Adaptive Bat Algorithm

Input: Population of bats $BA_x: (1 \leq x \leq a)$

Output: Best position of Bat BA^*

Start

Initialization

For $\forall a$

Calculate the level of loudness and pulse emission rate

While (w< Maximum iterations)

Create novel solutions by using equation (18)

If $(\boldsymbol{Rand} > \boldsymbol{E_x})$

{

Position update using random walk

}

End if

Develop new ideas with random flies

If $(\boldsymbol{Rand} < \boldsymbol{D_N}$ & ***Fitness*** $(\boldsymbol{B_*}) <$ ***Fitness*** $(\boldsymbol{B^*}))$

{

Increase the bat's location

Increment pulse emission rate E_x decrement loudness D_x

}

End if

Filter the response by its level of fitness

Calculate the best solution available at the moment

End While

Produce the best solution BA^*_{bax}

Stop

3.5 Hybrid CNN-LSTM for Image Classification

To improve picture classification accuracy, the Hybrid CNN-LSTM model combines Long Short-Term Memory (LSTM) for temporal analysis with Convolutional Neural Networks (CNN) for spatial feature extraction. CNN uses filters to map features, and then LSTM to grasp temporal environment. Overfitting is avoided via dropout, while network training is stabilized by batch normalization. Accurate classification is ensured by fully linked layers with SoftMax activation. This integrated method improves picture classification performance in a variety of applications by effectively identifying both temporal and spatial patterns.

Convolutional Neural Network: This model utilizes the advantages of CNN and LSTM layers to do thorough feature extraction. After using CNN to get spatial features via convolution, max pooling, and batch normalization, the output is processed by LSTM for

temporal characteristics, with dropout layers preventing overfitting. The dual-layer technique is repeatedly used three times, employing different neurons and filters. This culminates in a completely linked layer that utilizes SoftMax activation to achieve accurate categorization in various applications.

The convolution layer of CNN utilizes filters that slide over input matrices and calculate dot products to generate feature maps. Each dot product in the output matrix represents a scalar value obtained by sliding horizontally and vertically. Activation functions based on thresholds determine the firing of neurons. The Eq. (19) that follows after activation controls the output of the neuron, enabling the extraction of crucial characteristics in tasks involving image and pattern recognition

$$Z = a\left(\sum_{i}^{p\times q} w_i v_i + b\right) \tag{19}$$

The activation function is represented by a, weights are represented by w, input data is represented by v, bias is represented by b, and the sizes of the input data matrix are represented by p *and* q. The pooling layer follows the convolution operation on the data. Pooling layer serves to decrease dimensions of produced matrix to mitigate overfitting and improve the learning process. The Max pooling approach reduces the sample size while preserving the weights.

Batch Normalization: Batch Normalization (BN) is essential for maintaining deep neural networks by mitigating covariance changes among layers. The procedure enhances optimization, reduces generalization mistakes, and modifies the output of a CNN by normalizing the input data to a unit norm before processing in the LSTM layer. Formulas incorporate data (X) obtained by Max pooling, batch mean (μB), batch variance (δB), and ε to guarantee a denominator that is not equal to zero is expressed in Eq. (20)

$$\hat{X} = \frac{X - \mu_B}{\sqrt{\delta_B^2 + \in}} \tag{20}$$

The Eq. (21) result will be analyzed using variables γ *and* β. This procedure produces $\hat{Y}$, with γ and β trained for improved learning output in the learning process.

$$\hat{Y} = \gamma\hat{X} + \beta \tag{21}$$

Long-Short-Term Memory: The fundamental principle of LSTM is the utilization of memory cells to encode and retain inputs over some time. Gates, equipped with activation functions, carry out computations on these memory cells. The LSTM architecture has four distinct gates: the forget gate, the update gate, the tanh gate, and the output gate. Learning entails modifying weights and activation values to efficiently produce temporal characteristics between input and output data, hence improving the model's predicting skills.

In an LSTM network, the input and output vectors ($X(t)$) have identical dimensions. The forget gate, which merges the current input $X(t)$ with the prior concealed state $X(t-1)$, determines whether information to keep or delete. The process of generating output entails applying the sigmoid function and multiplying it with the prior cell state $C(t-1)$. The update gate utilizes the sigmoid and tanh functions to calculate $C(t)$ by integrating input information. The LSTM network output is the current hidden state h(t) generated by the resulting gates is expressed in Eq. (22).

$$C(t) = \sigma(b + U \times X(t) + W \times a(t-1)) \tag{22}$$

Dropout: This method randomly drops neurons each epoch during training. Deep neural networks must undergo this procedure to avoid overfitting, which limits their ability to recognize variables in fresh data.

Fully Connected Layer: The last layer processes extracted map features. Fully Connected (FC) neurons link to all below them. Softmax activation is its main categorization function. Input data becomes a one-dimensional layer that classifies and assigns output probabilities. This layer produces the final effect. The output layer ultimately generates the result indicating either normal or abnormal activity.

3.6 Feedback

If the output is in a normal activity, there is no subsequent procedure. However, if the output exhibits abnormal activity, it sends to the data storage with defined label. It gives the recommendations or feedback to the patient regarding the abnormal activity.

4 Result and Discussion

This work presents an optimized CNN-LSTM model for activity identification in lower limb prostheses. This context presents a comparison between the utilization of CEEMD. The Adapted Bat Algorithm is utilized to optimize the model and increase its performance by extracting relevant characteristics. This section compares the recommended method's efficacy with a number of currently used methods.

4.1 Performance Metrics

- **Accuracy:**

To determine a test's accuracy, the results from all the cases are examined and the percentage of true positives and negatives are determined.

$$Accuracy = \frac{TN + TP}{Tp + FP + TN + FN} \tag{23}$$

- **Precision:**

Precision refers to the proportion of true positives that were successfully classified.

$$Precision = \frac{TP}{TP + FP} \tag{24}$$

- **Recall:**

Recall shows the number of right hits and true positives that were recalled or identified.

$$Recall = \frac{TP}{TP + FN} \tag{25}$$

- **F_Score**

The F-Score number strikes a balance between ensuring that each class only has one kind of data item and precisely identifying every data piece.

$$F_Score = \frac{2(precision.Recall)}{Precision + Recall} \tag{26}$$

- **Specificity**

Specificity refers to the likelihood that a student will pass, assuming that it is affirmative.

$$Specificity = \frac{TN}{TN + FP} \tag{27}$$

- **FDR**

FDR is the "False Discovery Rate". It is a statistical measure that quantifies the ratio of falsely identified positive data points to the overall count of positively identified data points

$$FDR = \left(\frac{Fp}{(Fp + Tp)}\right) \tag{28}$$

- **FNR**

FNR denotes the "False-Negative Rate". It is defined as the percentage of incorrectly categorized cases that are mistakenly assigned a negative label in relation to the total number of instances that are assigned a positive label.

$$FNR = \left(\frac{Fn}{(Fn + Tp)}\right) \tag{28}$$

- **FPR**

FPR denotes the "False-Positive Rate". It may be defined as the ratio of the total amount of negative data to the percentage of positive data that were erroneously categorized.

$$FPR = \left(\frac{Fp}{(Fp + Tn)} \right) \tag{29}$$

- **MCC**

MCC refers to "Matthew's Correlation Coefficient". It is a statistical measure used to assess the quality of binary classification models. It is computed based on four values *Tp*, *Tn*, *Fp*, and *Fn*.

$$MCC = \left(\frac{((Tp \times Tn) - (Fp \times Fn))}{\sqrt{((Tp + Fp)(Tp + Fn)(Tn + Fp)(Tn + Fn))}} \right) \tag{30}$$

- **NPV**

NPV refers to the "negative predictive value". It can be described as the probability of accurately classifying genuine non-hit variants as non-hits, given a specific threshold.

$$NPV = \left(\frac{Tn}{(Tn + Fn)} \right) \tag{31}$$

4.2 Performance Evaluation

The performance measures of four distinct neural network models are being evaluated: CNN, DNN, LSTM, and Optimized CNN-LSTM. The measure of accuracy assesses the level of validity of the models in a comprehensive manner.

Table 3 provides a thorough comparison of four different models CNN, LSTM, DNN, and an Optimized CNN-LSTM using many performance indicators. The Optimized CNN-LSTM surpasses the individual models in most criteria, demonstrating its improved prediction ability. The Optimized CNN-LSTM model gets the best accuracy (0.99235) compared to all other models, demonstrating its higher overall predictive accuracy. The improved model has exceptional performance in reducing false positives and accurately collecting meaningful patterns, as indicated by the precision (0.98834), recall (0.97967), and F-score (0.99854) metrics. The Optimized CNN-LSTM has exceptional specificity (0.99781) and sensitivity (0.98452), indicating its proficiency in accurately identifying negative and positive occurrences, respectively. The Optimized CNN-LSTM model has a solid overall performance, as indicated by its highest MCC of 0.99367. The improved model demonstrates a high NPV of 0.99218, indicating its effectiveness in accurately forecasting negative situations. The proposed achieves a low FPR of 0.05798

Table 3. Performance analysis for proposed Vs existing classifier

	CNN	LSTM	DNN	Optimized CNN-LSTM
Accuracy	0.98745	0.96425	0.93214	0.99235
Precision	0.96347	0.97094	0.97	0.98834
Recall	0.95327	0.96785	0.95672	0.97967
F-score	0.95902	0.98753	0.96734	0.99854
Specificity	0.98642	0.97321	0.95786	0.99781
Sensitivity	0.97985	0.95467	0.96	0.98452
MCC	0.98745	0.94362	0.97643	0.99367
NPV	0.95897	0.96785	0.98756	0.99218
FPR	0.07943	0.06984	0.07999	0.05798
FNR	0.06904	0.05872	0.05128	0.04231

and a low FNR of 0.04231, demonstrating its effectiveness in minimizing both types of mistakes. Figure 2 shows the Analysis of classifier performance including the evaluation of precision, accuracy, F-score, recall, sensitivity, specificity, MCC, NPV, FPR, and FNR.

The values acquired for the suggested model are connected to the proposed model and the methods that are currently in use in [16] and [1]. The evaluation is conducted based on metrics such as accuracy, precision, sensitivity, and F-score. The comparison is presented in Table 4.

Table 4 presents a comparison of performance metrics of the current studies ([16] and [1]) with the suggested model. The metrics being compared are accuracy, precision, sensitivity, and F-score. The accuracy in [16] is 98.32%, the precision is 97.49%, and the sensitivity is unspecified. The results show that [1] has an accuracy of 98.1%, a precision of 86.7%, a sensitivity of 96.4%, and an F-score of 91.3%. On the other hand, the suggested model surpasses both, attaining superior values in all metrics: 99.235% accuracy, 98.834% precision, 98.452% sensitivity, and an amazing 99.854% F-score. This demonstrates the improved prediction capacity and well-rounded performance of the proposed model as compared to current research. Figure 3 explains the Analysis of classifier performance for the present model including the evaluation of accuracy, precision, sensitivity, and F-Score.

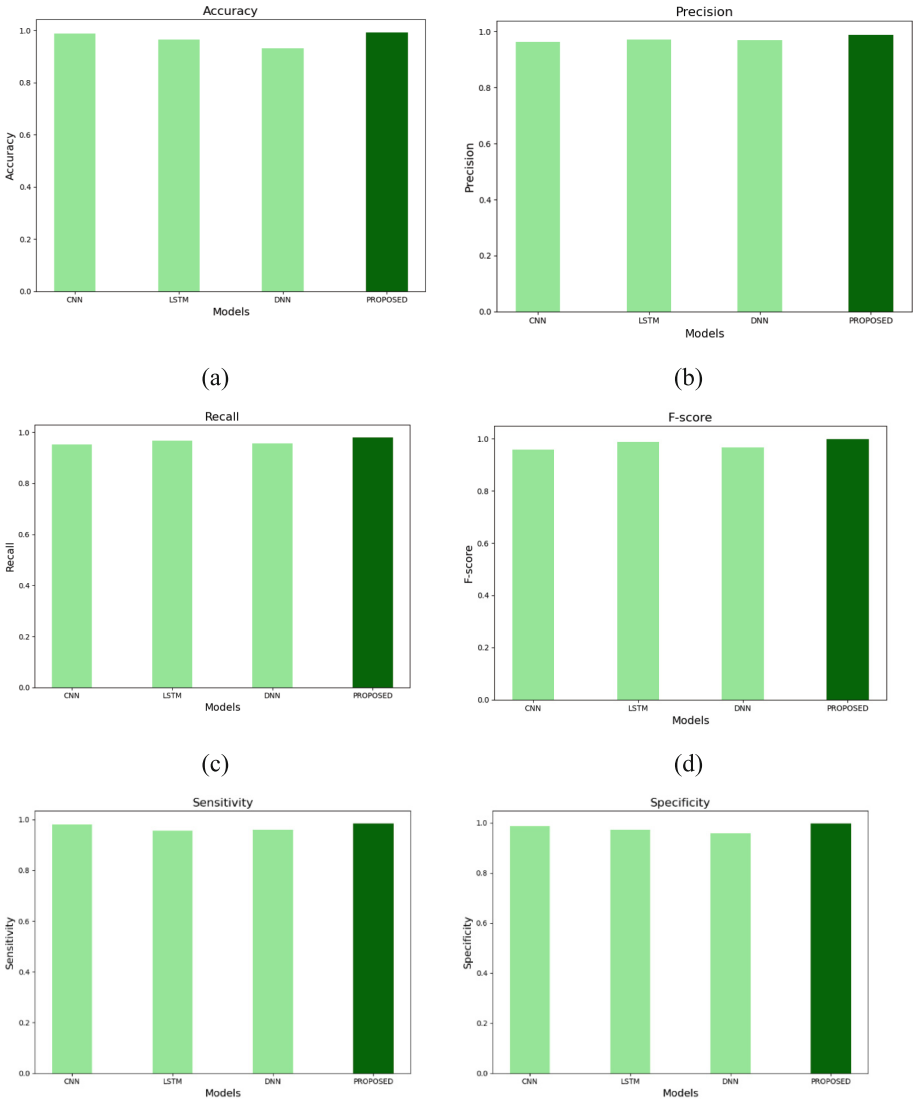


Fig. 2. Analysis of classifier performance includes the evaluation of (a) accuracy, (b) precision, (c) recall, (d) F-score, (e) specificity, (f) sensitivity, (g) MCC, (h) NPV, (i) FPR, and (j) FNR

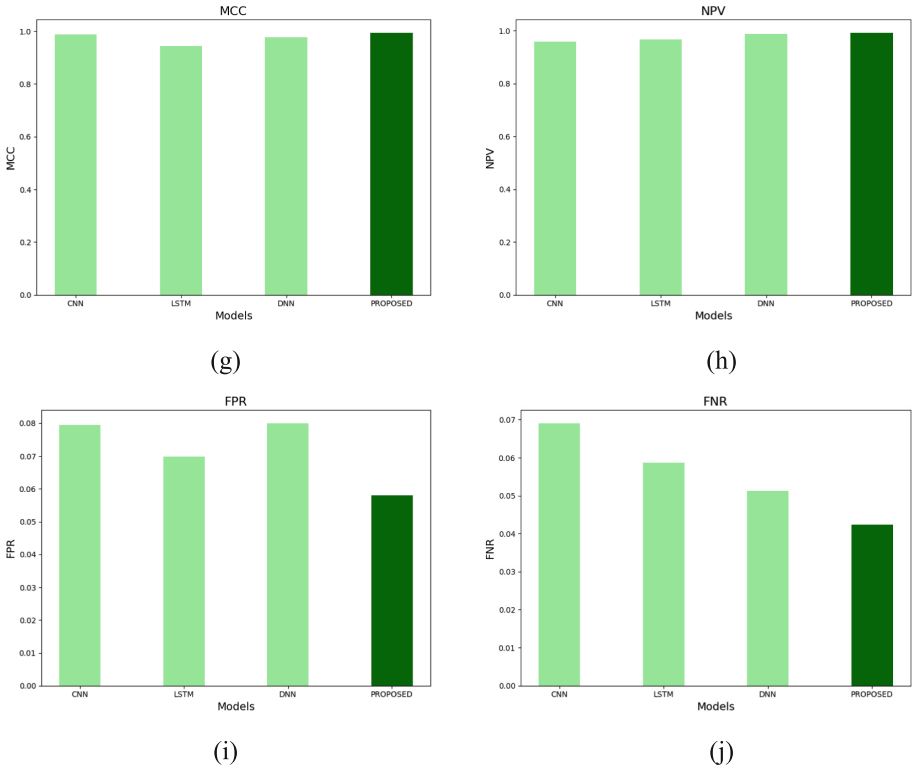


Fig. 2. (*continued*)

Table 4. Comparison of performance metrics for proposed model and existing techniques

Existing works	Accuracy	Precision	sensitivity	*F*-score
[16]	98.32	97.49	97.53	97.51
[1]	98.1	86.7	96.4	91.3
Proposed	99.235	98.834	98.452	99.854

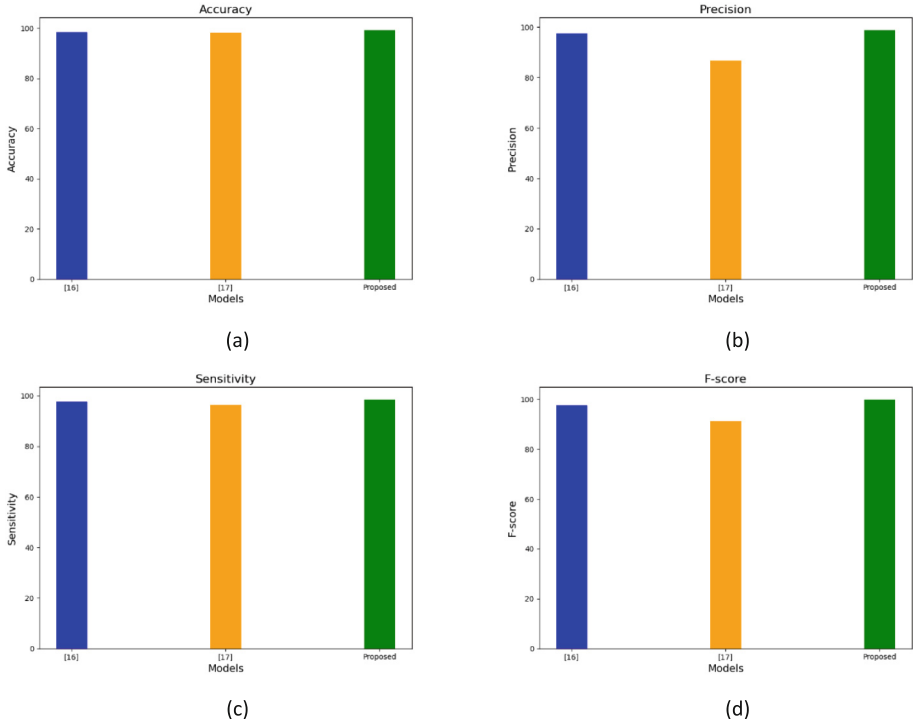


Fig. 3. Analysis of classifier performance for the present model includes the evaluation of (a) accuracy, (b) precision, (c) sensitivity, and (d) F-Score

5 Conclusion

Activity identification is essential for improving the operation of prosthetic devices by enabling their interaction with users' motions. However, many different data sources contribute to the complexity. To address the difficulties associated with integration, it is crucial to have a strong and effective approach. This work explores the profound influence of deep learning approaches on lower limb prosthesis, by greatly enhancing the detection of activities in the presence of complex data complexity. There were three main phases in the recommended process. The collected images underwent preprocessing using a Gaussian Bilateral Filter and Complete Ensemble Empirical mode decomposition. After the pre-processing step, features were retrieved from the processed data using a DTCWT. Retrieved features were later used in the feature selection procedure utilizing the Adaptive Bat Algorithm (BA). Characteristics were classified using an Optimized CNN-LSTM model. The Optimized CNN-LSTM produced greater percentages of accuracy (99.23%), precision (98.83%), recall (97.96%), sensitivity (98.45%), specificity (99.78%), F-score (99.85%), NCC (99.36%), and NPV (99.21%).

Future Work

By investigating supplementary sensor modalities, the system aims to improve its capacity to adjust to various user environments. Examining the practicality of implementing the model in real-time and addressing any possible constraints would enhance its usefulness for integrating it smoothly into the everyday routines of individuals using lower limb prostheses.

References

1. Slemenšek, J., et al.: Human gait activity recognition machine learning methods. Sensors **23**(2), 745 (2023)
2. Barberi, F., Anselmino, E., Mazzoni, A., Goldfarb, M., Micera, S.: Toward the development of user-centered neurointegrated lower limb prostheses. IEEE Rev. Biomed. Eng. (2023)
3. Hussain, T., Iqbal, N., Maqbool, H.F., Khan, M., Awad, M.I., Dehghani-Sanij, A.A.: Intent based recognition of walking and ramp activities for amputee using sEMG based lower limb prostheses. Biocybernet. Biomed. Eng. **40**(3), 1110–1123 (2020)
4. Xu, G., Wan, Q., Deng, W., Guo, T., Cheng, J.: Smart-Sleeve: a wearable textile pressure sensor array for human activity recognition. Sensors **22**(5), 1702 (2022)
5. Huaroto, J.J., Suárez, E., Vela, E.A.: Wearable mechatronic devices for upper-limb amputees. In: Control Theory in Biomedical Engineering, pp. 205–234. Academic Press (2020)
6. Lovegreen, W., Murphy, D.P., Stevens, P.M., Seo, Y.I., Webster, J.B.: Lower limb amputation and gait. In: Braddom's Physical Medicine and Rehabilitation, pp. 174–208. Elsevier (2021)
7. Meng, Z., et al.: Recent progress in sensing and computing techniques for human activity recognition and motion analysis. Electronics **9**(9), 1357 (2020)
8. Su, B.Y., Wang, J., Liu, S.Q., Sheng, M., Jiang, J., Xiang, K.: A CNN-based method for intent recognition using inertial measurement units and intelligent lower limb prosthesis. IEEE Trans. Neural Syst. Rehabil. Eng. **27**(5), 1032–1042 (2019)
9. Vijayvargiya, A., Gupta, V., Kumar, R., Dey, N., Tavares, J.M.R.: A hybrid WD-EEMD sEMG feature extraction technique for lower limb activity recognition. IEEE Sens. J. **21**(18), 20431–20439 (2021)
10. Qin, P., Shi, X.: Evaluation of feature extraction and classification for lower limb motion based on sEMG signal. Entropy **22**(8), 852 (2020)
11. McDonald, C.G., Sullivan, J.L., Dennis, T.A., O'Malley, M.K.: A myoelectric control interface for upper-limb robotic rehabilitation following spinal cord injury. IEEE Trans. Neural Syst. Rehabil. Eng. **28**(4), 978–987 (2020)
12. Semwal, V.B., Gupta, A., Lalwani, P.: An optimized hybrid deep learning model using ensemble learning approach for human walking activities recognition. J. Supercomput. **77**(11), 12256–12279 (2021)
13. Park, H., Kim, K., Shin, D., Shin, D.: BGP dataset-based malicious user activity detection using machine learning. Information **14**(9), 501 (2023)
14. Zhou, B., et al.: Accurate recognition of lower limb ambulation mode based on surface electromyography and motion data using machine learning. Comput. Methods Programs Biomed. **193**, 105486 (2020)
15. Barberi, F., Anselmino, E., Mazzoni, A., Goldfarb, M., Micera, S.: Toward the development of user-centered neurointegrated lower limb prostheses. IEEE Rev. Biomed. Eng. (2023)
16. Vijayvargiya, A., Singh, B., Kumar, R., Desai, U., Hemanth, J.: Hybrid deep learning approaches for sEMG signal-based lower limb activity recognition. Mathematical Problems in Engineering (2022)
17. Iqbal, N., Khan, T., Khan, M., Hussain, T., Hameed, T., Bukhari, S.A.C.: Neuromechanical signal-based parallel and scalable model for lower limb movement recognition. IEEE Sens. J. **21**(14), 16213–16221 (2021)
18. Peng, F., Zhang, C., Xu, B., Li, J., Wang, Z., Su, H.: Locomotion prediction for lower limb prostheses in complex environments via sEMG and inertial sensors. Complexity **2020**, 1–12 (2020)

19. Zhang, X., Tao, S.: Research on pattern recognition of lower limb motion based on convolutional neural network. Wirel. Commun. Mob. Comput. **2022**, 1–8 (2022)
20. Wang, Q.: Research on the improved CNN deep learning method for motion intention recognition of dynamic lower limb prosthesis. J. Healthcare Eng. (2021)



A Hybrid Approach to Cardiovascular Disease Detection: The Synergy of YOLOv9, CapsNet, and XGBoost

Gorapalli Srinivasa Rao and G. Muneeswari(✉)

School of Computer Science and Engineering, VIT-AP University, Amaravathi, Andhrapradesh, India
srinivasarao.22phd7042@vitap.ac.in, muneeswari.g@vitp.ac.in

Abstract. Cardiovascular Stroke is a potentially fatal condition that requires careful attention to achieve the best results. Early detection and intervention are crucial to reduce mortality and long-term damage associated with stroke. This study provides new solutions to these urgent needs by offering instant stroke detection based on deep learning and machine learning to increase performance. The most important goal of this research is to create a reliable and accurate system that can instantly distinguish strokes, allowing doctors to make informed decisions. Traditional stroke diagnosis technology relies on the interpretation of medical images, which is laborious and prone to human error. Although ML and DL approaches have demonstrated potential for automating this procedure, difficulties still exist since large and varied datasets are required. In order to overcome these obstacles, we suggest YOLOv9 + Capsnet in conjunction with XG-BOOST algorithms on large-scale datasets that include stroke and non-stroke cases according to the cardiovascular disease of the subjects seen in the pictures. Through this process of training, the model gains the ability to recognise complex patterns and features related to strokes, which improves the accuracy of its diagnosis. With the ability to increase patient caring and diagnosis of strokes, this optimised model has great promise for saving lives and raising the standard of healthcare. This article aims to identify the most effective and practical machine learning and deep learning algorithms for the prediction of cardiac illnesses by summarising some of the recent research in this area. Prediction systems' future directions have also been taken into account.

Keywords: Cardiovascular stroke · Machine Learning · Deep Learning · Capsule Network · XG-Boost · Yolov9

1 Introduction

The World Health Organisation predicts that by 2030, there would be a 24.5 million rise in deaths from cardiac hazard factors, which incorporates high blood pressure, being overweight, diabetes, finally smoking [1]. Worldwide, CVD is thought to be the central cause of mortality. The most common cardiovascular illnesses include coronary heart

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 256–265, 2025.
https://doi.org/10.1007/978-3-031-86296-0_18

disease, thrombosis of deep veins, peripheral arterial disease, cerebrovascular disease, strokes, pulmonary embolism, and transient ischemic episodes [2]. Its causes are frequently complex and include stress, eating unhealthy foods, leading a sedentary lifestyle, being overweight or obese, not exercising, having high BP, smoking, having diabetes, binge drinking alcohol, etc. More than 80% of deaths due to CVD are caused by strokes and heart attacks, and a startling one-third of those fatalities happen among people under the age of 70. There is a critical need for efficient CVD prevention, early detection, and enhanced care to decrease the burden of CVDs on public health. Early identification of heart strokes is therefore crucial since it allows researchers to evaluate and develop effective pharmacological and way of living interventions. This may support to stop or slow the development of CVD, reducing the imminence of death, which makes it especially important [3]. Users have embraced and implemented machine learning and deep learning, two well-known and well-liked AI idea variants, in a variety of facets of our daily life [4]. ML techniques, as opposed to naive ML and other approaches, are dependable and effective for predicting CVDs [5]. Deep learning techniques include more hidden layers and were trained on the original, vast amounts of data, increasing prediction accuracy [6]. The main intention of this proposal is to build a structured automated framework for the identification of cardiac conditions in image collections based on ML and DL. In this work we suggested, a novel idea for predicting cardiovascular strokes based on YOLO V9 + Capsule Network + XG-BOOST technique for increased forecast accuracy, increased productivity, and decreased personnel expenses in large general hospitals. The important contribution of this study is as follows:

- We examine recent studies on heart disease predictions in ML & DL from 2022 to 2024 in this survey study.
- We looked at the accuracy of the heart illness recognition systems to determine how well they performed.
- The main objective of problem-solving abilities is to employ ML and DL techniques in heart disease identification

This is how the rest of the article is structured. The relevant work based on diagnosing cardiovascular stroke identification using ML and DL strategies is represented in Sect. 2. Yolov9 advantages & Challenges discussed in Sect. 3. Later Capsule network and XG-BOOST information provided in Sect. 4. Then finally Results & Discussion along with Conclusion and Future Trends revealed in Sects. 5 and 6.

2 Related Work

Yanjun Li et al. suggested a preprocessing method as a deep learning precondition that combines image fusion and vessel improvement. Enhancing the fuzzy factors in coronary angiographic pictures increases the neural network's sensitivity to stenosis attributes. Our suggested procedure can enhance the efficiency of deep learning networks applications on the images from 6 common graphical image angles by evaluating five neural networks, including RFCN and YOLO-Inception-resnetv2. The results demonstrated that the suggested approach can be used to modify the deep model's recognition capacity for fine artery stenosis and is appropriate as a pretreatment strategy for deep learning-based coronary angiographic image processing [7]. Ya-Fang Chen et al. suggested a deep

learning architecture employing transfer learning to identify MRI scans inside the carotid artery for assessing the risk of stroke. We fine-tuned and modified the hyperparameters of pre-trained YOLO, Mobile Net, and RCN models in accordance with our problem. 265 patients' records from 2nd associated hospital of Fujian Medical University were gathered; of these, 116 individuals have a high hazard of developing plaques, while the rest 149 have constant health and a low imminence. Subsequently, we split the data into two groups: 30% for the testing set and 70% for training utilising a random selection method. In the assessment of stroke risk, proposed trained YOLO framework obtained 94.81% accuracy rate, RCNN secured 92.53% efficiency, and Mobile Network obtained 90.23% accuracy in recognising carotid plaque utilising MRI scans [8]. The suggested approach makes advantage of Better YOLO-v4. It broadens our understanding of small objects' semantics. The echocardiogram images are classified and features are extracted using an Enhanced YOLO-v4 with CSPDarkNet53. The capacity to predict cardiac disease was proven by integrating the results of both categorization systems. With 97.25% accuracy rate for regular sensor input and 98.87% accuracy-rate for irregular sensor input, the LBO-YOLO-v4 procedure identified both types of data. With the CSDarkNet53 technique, the enhanced YOLO-v4 that has been suggested provides superior categorization among echocardiogram images [9]. The suggested methodology shows how advanced machine learning, especially deep learning, might be useful in evaluating cardiovascular health. Based on the Arrhythmia dataset from MIT-BIH. The study surpasses present state-of-the-art techniques with an moderate success rate of 98% [10]. In this article, an efficient deep learning-based system for predicting cardiovascular illness from patient data is proposed, namely a CNN incorporating bidirectional LSTM and this technique produced promising experimental results. With 94.507% accuracy [11]. The suggested approach is a multi-step process. To remove noise from the ECG raw data, a low pass filter is utilised in the first stages along with a R–R peak extraction. A fully connected layers design based on convolutional neural networks and employing various network optimizers is the suggested second stage. For validation, several ECG datasets have been utilised. For complexity analysis, CPU and GPU are used to implement the entire system. For testing, training, and validation, the classification accuracy values for the anticipated upgraded PTB dataset are 99.37%, 99.15%, and 99.31%, respectively. Furthermore, 99.5%, 99.06%, and 99.34%, respectively, are the training, verification, and assessment accuracy levels for the MIT-BIH database [12]. To enhance the identification accuracy rates of heart illness, seven deep learning and machine learning classifiers were implemented. Insights for creating reliable models of prediction for myocardial infarction are gained from this study's examination of several classifiers' performances. The study's conclusions highlight how well an XG-Boost model may be precisely tuned to treat cardiovascular disorders. 98.71% F1 score, 99.14% preciseness, 98.29% recall, and 98.50% accuracy are the impressive results of this optimisation [13]. This research presents a k-modes clustering method that can increase classification accuracy using some designs like XGBoost (XGB), random forest (RF), a Decision-Tree classifiers (DT), and multilayer perceptron (MP). To maximise the results, GridSearch-CV was utilised to hypertune the employed model's attributes. Real-world set of 70,000 cases from Kaggle, the suggested framework is used. After training session on 80–20 splitting of data, the models' accuracy rate was observed like: The decision tree showed random

forest showed 87.05% with cross validation and 86.92% without; and the multilayer perceptron showed 87.28% with and 86.94% without cross validation, 86.37% with cross validation and 86.53% without; XGBoost showed 86.87% with cross validation and 87.02% without [14]. Using the Cleveland Heart Disease Dataset, the suggested model's effectiveness is assessed and contrasted with Random Forest and Extra Tree classifiers. For the purpose of evaluating performance, five distinct assessment measures were employed: F1-score, sensitivity rate, accuracy-rate, specificity rate and area under the curve of ROC charts. The experiment's results demonstrated its suitability and efficiency in heart illness prediction. Furthermore, the suggested framework presents better than the primary models that were proposed. Furthermore, our suggested strategy has a more prediction success rate of 91.8% [15].

3 YoloV9 Advantages and Challenges

YOLOv9 differentiates itself in the pursuit of ideal regional-time object identification by taking a unique strategy to resolve the information loss issues that come with deep neural networks. YOLOv9 achieves accuracy and performance by combining PGI with the updated GELAN structure. This not only enhances the learning potential of the framework, but also ensures that important information is preserved during research.

Solving the problems caused by data loss in deep neural networks is the main focus of the YOLOv9 invention. Its design is based on the knowledge of the principle and practical application of the recovery function to ensure that YOLOv9 maintains its accuracy rate and efficiency level.

The Data Bottleneck Rule

The Data Bottleneck Rule describes a fundamental problem in deep learning: As data moves through higher levels of the network, data loss increases. This is expressed mathematically as:

$$h_\Theta(A) \geq x(A, A) \geq x(A, k_\pi(f_\Theta(A)))$$

Of these, h and k represent the transfer function with Θ and π, respectively, and x represents the shared data. YOLOv9 overcomes this problem by using Programmable Gradient Information to help manage important information deep in the network. This assures reliable gradient formation in the aggregate, thereby improving frame convergence and performance efficiency.

Reversible-Operations

The framework of YOLOv9 is also based on the recovery concept, here reversible functions are methods that can be reversed without any data loss, as mentioned below:

$$X = v_z(r_\Psi(X))$$

where z and Ψ stand for the reversible and inverse functions' respective properties. This feature is crucial for learning deep networks because it gives the system's data-flow and enables precise adjustments to the framework's properties. An inverse function is

incorporated into the design of YOLOv9 to lessen the likelihood of inaccurate data, particularly in the layers, making the data more valuable to the company storing the search objects. Particular attention should be made to information loss since there is little optimisation and crucial information is readily lost throughout the feedforward process. By utilising PGI and reversible procedures, YOLOv9's architecture makes sure that crucial information is properly stored and utilised to recognise correct goods, even in basic frameworks.

Gradient Information That Is Programmable (PGI)
To solve the issue of physical data and ensure the preservation of important data in all deep procedures, PGI is a revolutionary topic first proposed in YOLOv9. This makes it possible to implement reliable gradients, thus simplifying framework updates and enhancing overall performance.

Potential Benefits of YOLOv9 (Fig. 1)



Fig. 1. Yolov9 Advantages

YOLOv9's Speculative Challenges (Fig. 2)

Even though YOLOv9 is still in the early stages of development, its ability to enhance the number of searches, recovery, speed, and detection accuracy builds on the general trend in improvements observed in the previous YOLO releases. However, these improvements also come with some drawbacks, such as higher operational costs, high complex framework training, and the need for large & high-quality data. These types of issues need to be resolved before YOLOv9 can be deployed and utilized in many applications.

4 Capsule Network and XG-BOOST

With CapsNets, the idea of "capsules"—groups of neurons that stand in for different attributes of objects or sections of objects—is introduced. A capsule network's essential elements are as follows:

Vectors known as capsules contain the instantiation parameters that determine particular characteristics, like scale, position, and orientation. Every capsule produces a

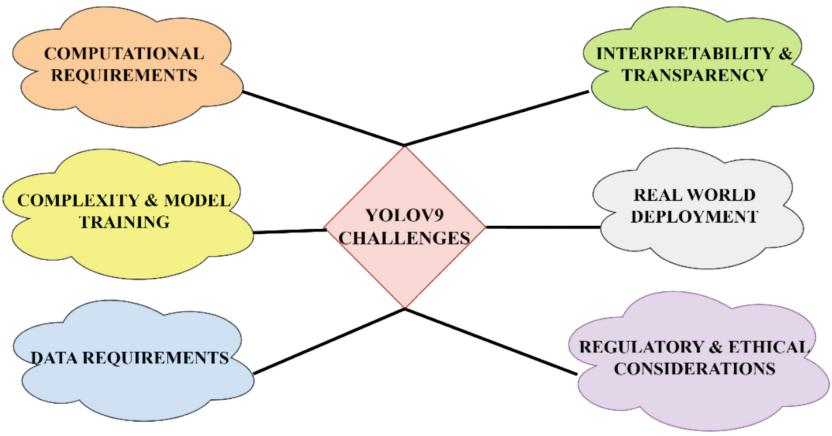


Fig. 2. Yolov9 Challenges

vector, the orientation of which indicates the feature's attributes and the length of which indicates the likelihood that that attribute is present.

The output from layers of convolution is directly received by primary capsules, which are the primary layer of capsules. They convert the maps of convolutional features into capsule form.

Capsules at higher levels depict more intricate details and entities. They get inputs from capsules at a lower level by a mechanism called dynamic routing.

An essential Caps-Nets mechanism is dynamic routing, which iteratively modifies the coupling coefficients among capsules according to the degree of agreement across lower- and higher-level capsules.

Extreme(X)-gradient boosting is an ML approach that has shown great promise in the identification of cardiovascular illness from image datasets. XG-Boost specializes in handling large and complex data using its robust and reliable tree optimization algorithm, making it specifically designed for medical applications. XG-Boost can diagnose heart disease by analysing data such as echocardiograms, CT scans, and MRI images, extracting complex patterns and features that indicate disease. Effective models are accurate and reliable because they have the ability to control for missing variables and the integration of orderly processes.

Additionally, XG-Boost's parallel processing capability minimizes learning time, allowing rapid development and use of predictive models. By integrating clinical data and imaging, XG-Boost aids risk stratification, treatment planning, and diagnostic accuracy to improve the outcomes of cardiovascular care.

YOLOv9, Caps-Net, and XG-Boost together provide a potent hybrid method for detecting cardiovascular disease:

- In imaging data, YOLOv9 is utilised to identify and locate cardiovascular structures and possible anomalies.
- Spatial hierarchies and interactions within the identified regions are captured using capsule networks, which yield a reliable feature representation for additional research.

– To predict the risk and outcomes of cardiovascular disease, clinical data and the features retrieved by YOLOv9 and Caps Net are incorporated into an XG-Boost model.

With its great accuracy and speed, YOLOv9 is a huge leap in real-time object detection. A potent hybrid model for the identification of cardiovascular disease may result from working together with Caps-Net and XG-Boost, which will take advantage of deep learning's advantages in extracting features and machine learning's in predictions and risk assessment. This combination has the potential to improve patient outcomes and decision-making by giving clinicians comprehensive, understandable insights. The proposed architecture shown below in Fig. 3.

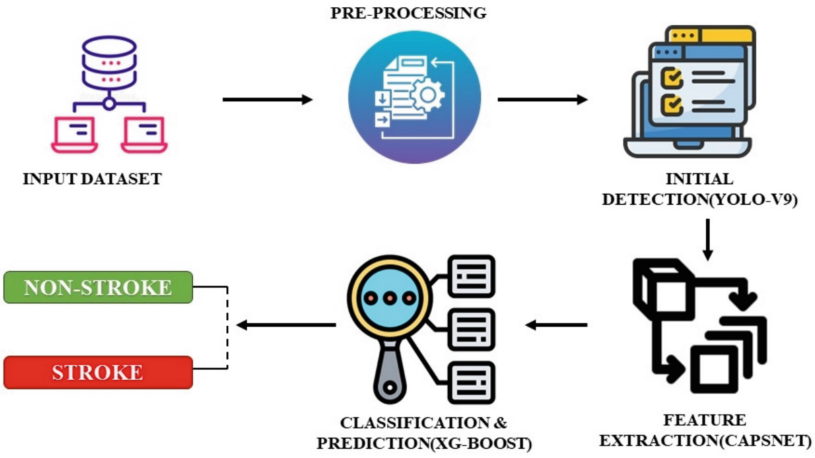


Fig. 3. Proposed Architecture

Challenges

Data Quality: Data quality refers to providing varied and high-quality data for validation and training.
Model Integration: YOLOv9, CapsNet, XGBoost, and other components should be integrated seamlessly, and their computational needs should be managed.
Interpretability: The capacity to interpret the model while keeping in mind the necessity of actionable and interpretable forecasts.

5 Results and Discussion

The writing audit states that experts from all over the world are starting to concentrate on the applications of deep learning and machine learning in healthcare, particularly in the area of illness prediction.

A certain number of exploration projects were considered between 2022 and 2024, and each project's findings were evaluated and looked into. The intricacy of deep learning and machine learning actually considerably reduces the workload for medical practitioners. Table 1 presents a comparative examination of cardiovascular illness utilising deep learning and machine learning.

Table 1. Comparative Study of Cardiovascular Disease Using Deep Learning and Machine Learning approaches.

Ref's	Techniques Used	Dataset	No.of Instances	Accuracy
[7]	YOLOv4 and R-FCN	Hokkaido University Hospital	250 Representative Images	75%
[8]	YOLOv3	Hospital of Fujian Medical University	265 Patients MRI images	94.81%
[9]	YOLO-v4 with CSPDarkNet53	Real Time Information	NA	98.87%
[10]	CNN + K-MEANS + MULTI-MODAL FUSION	MIT-BIH Dataset	100 Human Records	98%
[11]	CNN + BI-LSTM	UCI	76 Variables	94.50%
[12]	CNN	MIT-BIH, PTB	109,446 + 14,552 samples	99.50%
[13]	ML + DL Models	MENDLEY & KAGGLE	1303 samples	98.50%
[14]	MULTILAYER PERCEPTRON + XG-BOOST	CVD Dataset	70,000 patient records	87.28%
[15]	OPTIMIZED XG-BOOST	Cleveland Dataset	303 Instances	91.80%

Each author employed a few datasets in the literature review based on the available data. When compared to existing methodologies, the proposed innovative hybrid model will perform better in terms of prediction. It combines the capabilities of XG-Boost for reliable prediction and risk assessment, Caps-Net for deep extraction and classification of features, and YOLOv9 for real-time cardiovascular picture detection. The model also makes use of bulk datasets like the National Institutes of Health (NIH) Clinical Centre Chest X-ray Dataset (ChestX-ray14), UK Bio-bank, Cardiac MRI Datasets from Kaggle, Statistical atlases—computational frameworks of the Heart Dataset information, and real-time information from various medical organisations to improve the performance efficiency (Fig. 4).

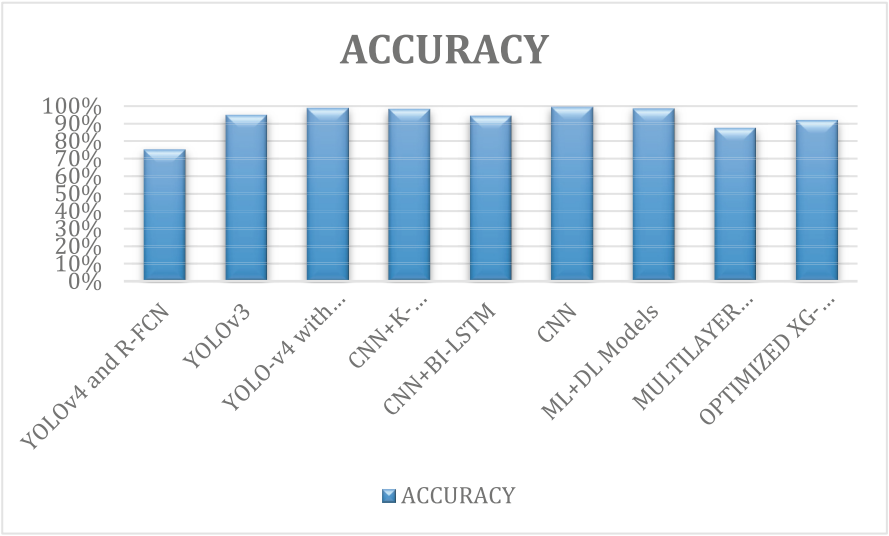


Fig. 4. Sophisticated analytical powers

6 Conclusion and Future Scope

By combining machine learning and deep learning, our research work has developed a novel, real-time stroke detection system that provides an effective and considerate solution. Together with Caps-net and XG-BOOST, we have created a system that has the potential to remarkably increase the timeliness and accuracy rates of stroke identification by integrating the sophisticated analytical powers of deep learning, namely through the application of YOLOv9 models. This method overcomes the drawbacks of conventional stroke detection techniques, including the need for large, varied datasets and their reliance on laborious, error-prone manual interpretation. The findings have far-reaching consequences, providing a mechanism to improve patient outcomes by empowering medical practitioners to make decisions more quickly and intelligently.

In the future, effectiveness and classification predictions for cardiovascular illness may be improved by combining the suggested methodology with some optimisation techniques and methods for selecting features based on real-time data from several organisations gathered with the aid of IoT devices.

References

1. Shrivastava, P.K., Sharma, M., Kumar, A.: HCBiLSTM: a hybrid model for predicting heart disease using CNN and BiLSTM algorithms. Meas. Sensors **25**, 100657 (2023)
2. Hossain, M.M., et al.: Cardiovascular disease identification using a hybrid CNN-LSTM model with explainable AI. Inform. Med. Unlock. **42**, 101370 (2023)
3. Al Reshan, M.S., Amin, S., Ali Zeb, M., Sulaiman, A., Alshahrani, H., Shaikh, A.: A robust heart disease prediction system using hybrid deep neural networks. IEEE Access (2023)

4. Moshawrab, M., Adda, M., Bouzouane, A., Ibrahim, H., Raad, A.: Reviewing multimodal machine learning and its use in cardiovascular diseases detection. Electronics **12**(7), 1558 (2023)
5. Taylan, O., Alkabaa, A.S., Alqabbaa, H.S., Pamukçu, E., Leiva, V.: Early prediction in classification of cardiovascular diseases with machine learning, neuro-fuzzy and statistical methods. Biology **12**(1), 117 (2023)
6. Malibari, A.A.: An efficient IoT-Artificial intelligence-based disease prediction using lightweight CNN in healthcare system. Meas. Sensors **26**, 100695 (2023)
7. Li, Y., Yoshimura, T., Horima, Y., Sugimori, H.: A preprocessing method for coronary artery stenosis detection based on deep learning. Algorithms **17**(3), 119 (2024)
8. Chen, Y.-F., et al.: Stroke risk study based on deep learning-based magnetic resonance imaging carotid plaque automatic segmentation algorithm. Front. Cardiovasc. Med. **10**, 1101765 (2023)
9. Alamelu, V., Thilagamani, S.: Lion based butterfly optimization with improved YOLO-v4 for heart disease prediction using IoMT. Inf. Technol. Control **51**(4), 692–703 (2022)
10. Singh, N., Gunjan, V.K., Shaik, F., Roy, S.: Detection of cardiovascular abnormalities using gradient descent optimization and CNN. Health Technol. **14**(1), 155–168 (2024)
11. Ahmad, S., Asghar, M.Z., Mazaed Alotaibi, F., Alotaibi, Y.D.: Diagnosis of cardiovascular disease using deep learning technique. Soft Comput. **27**(13), 8971–8990 (2023)
12. Fradi, M., Khriji, L., Machhout, M.: Real-time arrhythmia heart disease detection system using CNN architecture based various optimizers-networks. Multimedia Tools Appl. **81**(29), 41711–41732 (2022)
13. Ogunpola, A., Saeed, F., Basurra, S., Albarrak, A.M., Qasem, S.N.: Machine learning-based predictive models for detection of cardiovascular diseases. Diagnostics **14**(2), 144 (2024)
14. Bhatt, C.M., Patel, P., Ghetia, T., Mazzeo, P.L.: Effective heart disease prediction using machine learning techniques. Algorithms **16**(2), 88 (2023)
15. Budholiya, K., Shrivastava, S.K., Sharma, V.: An optimized XGBoost based diagnostic system for effective prediction of heart disease. J. King Saud Univ. Comput. Inf. Sci. **34**(7), 4514–4523 (2022)



Event Detection and Analysis from Social Media Data Using N-gram and Distil-BERT Model

Dharmendra Mangal[1,2(✉)] and Hemant Makwana[1]

[1] IET, Devi Ahilya University, Indore, India
mangaldharmendra83@gmail.com
[2] Medi-Caps University, Indore, India

Abstract. Now-days social media became popular source of communication. An event is an incident occurred in real-world. The Social media may be used to sense around the real world events as most of them are communicated on this platform. The sensed data are further used to detect significant events. This detected information will help the people and organization to act accordingly. Some events like natural disasters, accidents etc. require detection in real time as well as prompt action. In this work, we applied Distill-BERT model for event detection. Distill-BERT model is 60% faster and 40% smaller in size as compared to BERT-base model. This makes the event detection faster which is desired in real time processing. Also accuracy is not compromised as compared to base model (only ~3% compromised). This compromised is covered by applying N-gram model that improvise the system performance. The experimental results shows that F1-measure for BERT-base, Distill-BERT are 0.75 and 0.72 respectively. The proposed method improves the F1-measure to the value 0.76. Hence proposed approach work better for event detection especially in real time scenario.

Keywords: Social network service · Event Detection · BERT · Distill-BERT

1 Introduction

There are billions of users of social media from all around the world. Social media was first created to facilitate internet communication with friends and family. It was used to share the everyday information, happenings and news among friends and family [1, 2]. These were considered as private events the birthdays, marriages, and vacations are some few examples of the personal events. Individuals and businesses began utilizing social media to reach more clients once it was found that social media was quite useful. Nowadays, the social media becomes connecting media for people all around the globe. The influence of social media is worldwide. Various social media platforms are available line facebook, Instagram, X (earlier-Twitter) and YouTube etc. [3, 4].

If we define the term ‘event’ it simply an occurrence of change with respect to time and location. This means that time and locations are two important factors with respect to events. To detect any event, the happening information of the event is required. These information must be sensed by a sensor like temperature rising and falling is sensed using

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 266–276, 2025.
https://doi.org/10.1007/978-3-031-86296-0_19

thermometer. The Information communicated on social media can be used to sense the events which not always the private ones. The public events are needed to be detected as they may affect the life of the people in large [5]. The easy availability of multimedia tools raises the event sharing in the form of text, image and video massages. The events can be categorized in many ways as shown in Fig. 1.

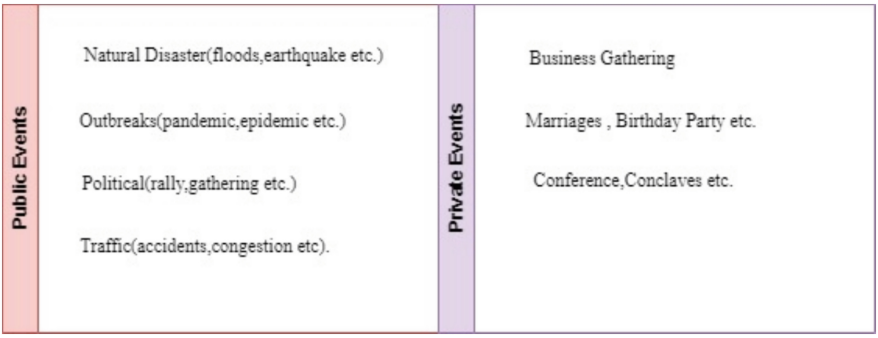


Fig. 1. Event Classification

Basically there are three approaches to solve event detection problems: supervised approach, unsupervised and semi-supervised. There are many supervised approaches like decision tree, Naïve Bayes classification, Support vector machine, CRF-based and others [6, 7]. While unsupervised approach includes clustering methods, graph based methods and others. The supervised approaches are more costly than unsupervised one as they required labeled dataset. Semi-supervised approaches follows middle path as the training of the model performed using small labeled data and large unlabeled data making it cost-balanced [8].

As usage of Internet and related technologies are increasing day by day led to growing the research in event detection significantly. Many methods have been proposed for event detection. These methods primarily are based on supervised and unsupervised learning [9]. Also need of benchmark datasets arises. Currently very few handful datasets are available for event detection research. Table 1 shows comparison among some popular datasets.

The most of the datasets used in research used are Wikipedia text based. The text messages collected from popular social media platform like X (earlier Twitter) are also being used in many research of event detection and analysis. Many researchers have even used live Twitter (now X) streams to detect events in real time which is very desirables in disastrous events like floods, earthquakes etc. Event detection is majorly a natural language processing (NLP) based problem. Named entity Recognition (NER) is a sub field of NLP set. The NER methods can be supervised and unsupervised. NER approaches are based on recognizing the desired entities like location, persons and actions. Most of the earlier approaches of event detection are NER based. Since the inception of the transformers concepts the NLP tasks are more efficiently solved using the transformers. But using transformer on text data directly may not necessarily lead to high accuracy. The data preprocessing and feature extraction tasks play very important role [10].

Table 1. Some popular Event datasets

Dataset	Description
EVents In News (EVIN 2014)	• English language • News corpus based • Covers 24,438 events with 453 event types
Event Wiki (2018)	• English language • Wikipedia based • Covers 21,275 events with 94 event types
Event KG (2020)	• Multilingual • DBPedia and YAGO • Covers 120,0000 events
MAssive eVENt (MAVEN 2020)	• English language • Wikipedia and Framenet based • Covers 111,611 events with 164 event types

Many studies have been conducted for detection of events from social media text. In order to get information about disaster events from crisis tweets on Twitter, Basu et al. employed techniques of data collecting, independent word embedding, and a combination of term and character word embedding. They used earthquake risk datasets from Nepal and Italy [11]. A social framework was created by Purohit et al. to categories and rank community service requests that are submitted on social media. By combining application domains and applying the linguistic clustering technique, they reduced redundancy and saved time. They consider data regarding the earthquake in Nepal, Hurricane in USA, and the floods in Alberta [12]. After the inception of transformer concepts in NLP arena there was tremendous shift. Instead machine learning models, transformer based models like BERT is used popularly. A transformer based on classification, CrisisBERT was recently suggested by Liu et al. It works better and is more stable than standard linear and DL techniques. The CrisisBERT technique was applied for event detection and disaster identification. Unfortunately, several features are now less effective due to the pre-trained transformer used in modern classification algorithms, which has negatively impacted the model's performance [13]. The presented research focuses on these tasks along with the fast transformer named as Distill-BERT which is a BERT model with knowledge distillation. BERT (Bidirectional Encoder Representation from Transformers) model is a very popular and widely applied transformer for NLP base problem solving. The BERT model is proposed by GoogleAI researchers in 2018. It was a revolutionary idea in the NLP domain. As basic BERT model is little slower compressed model is proposed by Hugging Face researchers named as DistilBERT [14, 15].

2 Methodology

The method we use is shown in Fig. 2.



Fig. 2. Methodology

2.1 Preprocessing

The preprocessing of the training text data plays very crucial role to improve the overall performance of the methods [16]. There are many activities comprises the text data pre-processing. In this proposed methodology we applied primarily the set of activities like dealing with emojis and emoticons, expanding chat-words, dealing with tags and special characters and stop word removal. Along with these we perform spelling correction so that feature extraction of the given text becomes smoother more.

2.2 N-gram Extraction

N-gram model is a probabilistic model of a language. Formal grammar model like regular and context free are hard binary models for legal sentences in a natural language. An N-gram model uses n − 1 words of prior context [17, 18]. This model is based on Markov presumption that the future behavior of a dynamic system depends on its recent states. N-gram model is used in many ways. In this work this model is used for feature extraction of a sentence to get event words occurrences. We obtain bi-grams (2-g) and trigrams (3-g) from text dataset.

Any text message (M) is a sequence of n words.

$M = W_1^n$ $(W_1 \ldots\ldots W_n)$ where W_i is i^{th} word in text message.

Unigram, Bigrams and Trigrams conditional probabilities can be estimated from the text message based on the relative frequency of word sequences. The following equations are used to calculate these probabilities:

$$P(W_i) = cnt(W_i)/cnt(M) \quad (1 - gram/unigram) \tag{1}$$

$$P(W_i|W_{i-1}) = cnt(W_{i-1}W_i)/cnt(M) \quad (2 - gram/unigram) \tag{2}$$

$$P(W_i|W_{i-2}W_{i-1}) = cnt(W_{i-2}W_{i-1}W_i)/cnt(M) \quad (3 - gram/trigram) \tag{3}$$

where $P(W_i)$ is the unigram probability and cnt(M) is the count of word W_i i.e. frequency of the word in the dataset. $P(W_i|W_{i-1})$ is the probability of word sequence W_iW_{i-1}. $P(W_i|W_{i-2}\ W_{i-1})$ is the probability of word sequence $W_{i-2}\ W_{i-1}W_i$. The cnt(M) = 4342 for non-disaster message and cnt(M) = 3271 for disaster messages as in the dataset we have 4342 non-disaster messages and 3271 disaster messages.

2.3 Applying Distill BERT

Distill-BERT is a lighter and faster BERT implementation based on knowledge distillation (shown in Fig. 3). The model has almost half number of layers as compared with the base model. Basically BERT is a deep learning model being trained on huge corpora [19]. Therefore, it requires much amount of time make it expensive on resource constrained environments like event detection in real time. Hence we used lighter flavor of BERT i.e. Distill-BERT. The Distill-BERT model is Diluted version of actual BERT model without much sacrificing the performance of the later one [20]. The Distill-BERT model is based on the concept of knowledge distillation. The knowledge distillation process is explained in the following figure.

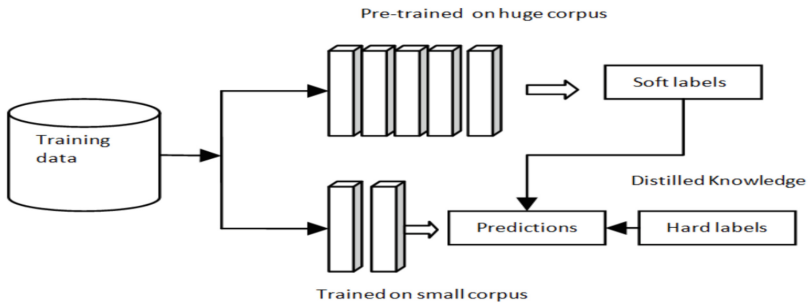


Fig. 3. Knowledge Distillation Process

Knowledge distillation is also knows as teacher-student learning. In this the student model (smaller model) is trained using the teacher model (larger model). Hence student model is the compressed model which makes it lighter and faster [21, 22]. The supervised learning based models are trained to predict the class based on maximizing the probability of estimated label. Usually the training objective comprises the reducing the cross entropy between predicted value distribution and on hot empirical distributions of the class labels during the training. The model is said to be trained properly if it predicts the correct class with a high probability and almost zero probability for other class [23, 24]. In knowledge distillation the student model is trained with cross entropy on soft labels which are passed from teacher model. Consequently the training loss is redefined as in Eq. (4).

$$\text{Loss} = -\Sigma_i T_i * \log(S_i) \tag{4}$$

where T_i is probability estimated by the teacher model while S_i is the probability estimated by the student model. This loss value is very significant for learning of student model as single example is tells more than single hard label value [25, 26].

3 Outcome and Inference

In this work we used datasets of tweets comprises thousands of text messages. This dataset was created by the company figure-eight and originally shared on their 'Data for Everyone' website. The dataset is divided into 3 parts: 70% part is used for training while 15% part is used for fine tuning, remaining 15% is used for testing purpose.

After applying set of preprocessing activities N-gram probabilities are calculated. The following tables shows the calculated unigram, bigram and trigram probabilities of frequent word sequence existed in training dataset. The different table are shown for alarming and non-alarming messages i.e. alarming messages contains event burst information. Tables 2, 3, and 4 show the unigram, bigram and trigram probabilities of most frequent words sequences respectively.

Table 2. (a) and (b): Unigram Probabilities of frequent word sequences alarming and non-alarming messages

Alarming Messages		
Word	**frequency**	**Probability**
fire	272	0.083154998
news	165	0.05044329
kill	162	0.049526139
bomb	150	0.045857536
year	150	0.045857536
amp	132	0.040354632
get	122	0.037297463
disaster	122	0.037297463
via	110	0.03362886
california	108	0.033017426
flood	104	0.031794558
suiscide	104	0.031794558
crash	104	0.031794558
police	104	0.031794558

Non alarming Messages		
Word	**frequency**	**Probability**
get	325	0.074850299
like	285	0.065637955
go	275	0.063334869
amp	262	0.060340857
new	180	0.04145555
one	140	0.032243206
would	138	0.031782589
make	130	0.02994012
want	125	0.028788577
love	120	0.027637034
body	120	0.027637034
see	110	0.025333947

Table 3. (a) and (b): Bigram Probabilities of frequent word sequences alarming and non-alarming messages

Alarming Messages		
Word	**frequency**	**Probability**
cheeky playful	68	0.02078875
stick cheeky	67	0.02048303
playful blowing	62	0.01895445
blowing raspberry	61	0.01864873
suicide bomber	60	0.01834301
northern california	41	0.01253439
oil spill	38	0.01161724
california wildfire	35	0.01070009
happy face	33	0.01008866
bomber detonate	30	0.00917151
suicide bomb	30	0.00917151

Non alarming Messages		
Word	**frequency**	**Probability**
body bag	70	0.0161216
laugh loud	45	0.01036389
crom body	38	0.00875173
happy face	35	0.0080608
look like	35	0.0080608
like youtube	34	0.00783049
stick cheeky	33	0.00760018
cheeky playful	33	0.00760018
playful blowing	32	0.00736988
face smiley	31	0.00713957
blowing raspberry	29	0.00667895

Table 4. (a) and (b): Trigram Probabilities of frequent word sequences alarming and non-alarming messages

Alarming Events

Word	frequency	Probability
stick cheeky playful	68	0.02078875
cheeky playful blowing	62	0.018954448
playful blowing raspberry	61	0.018648731
suicide bomber detonate	40	0.012228676
northern california wildfire	39	0.011922959
late home raze	38	0.011617242
pak suicide bomber	38	0.011617242
home raze northern	38	0.011617242
bomber detonate bomb	38	0.011617242
raze northern california	37	0.011311526

Non alarming Events

Word	frequency	Probability
stick cheeky playful	34	0.007830493
cheeky playful blowing	33	0.007600184
happy face smiley	31	0.007139567
cross body bag	29	0.00667895
reddit quarantine offensive	24	0.005527407
quarantine offensive context	19	0.004375864
pick fanaury	18	0.004145555
reddit new context	17	0.003915246
china stock market	16	0.003684938

The following figure shows how fine tuning of the model is done well. The development part of the dataset is used for model parameter adjustments while training. This improves the models performance metrics like accuracy, loss value and F1-score [27] (Figs. 4, 5, and 6).

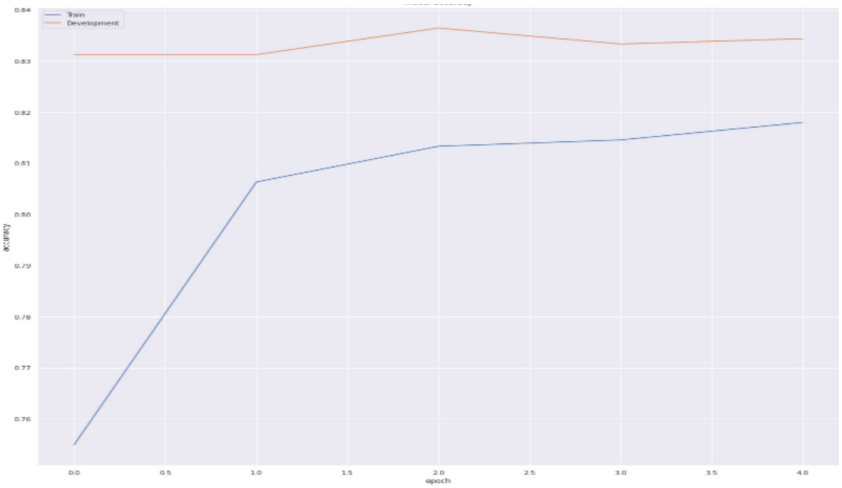


Fig. 4. Fine tuning in multiple epochs – accuracy metric

Accuracy metric is more significant when we are more interested in true positive and true negative while F1-score metric signifies false positive and false negative [28]. The following figure shows the confusion matrix obtained (Fig. 7).

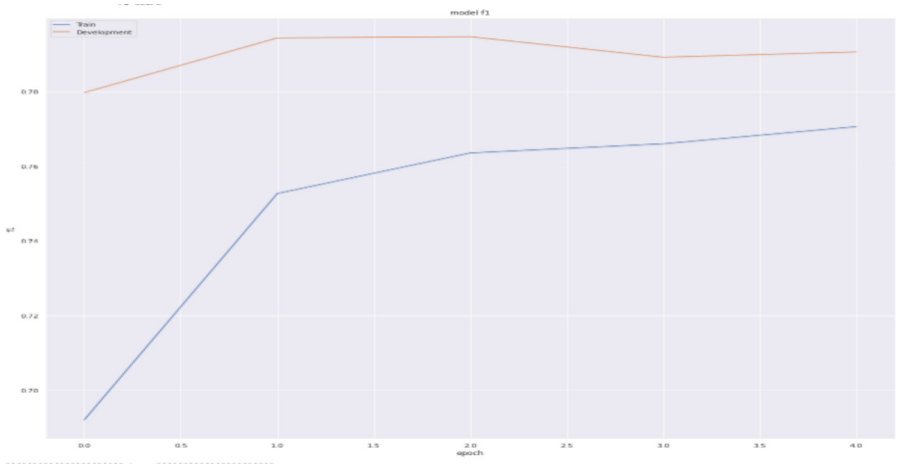


Fig. 5. Fine tuning in multiple epochs – F1 metric

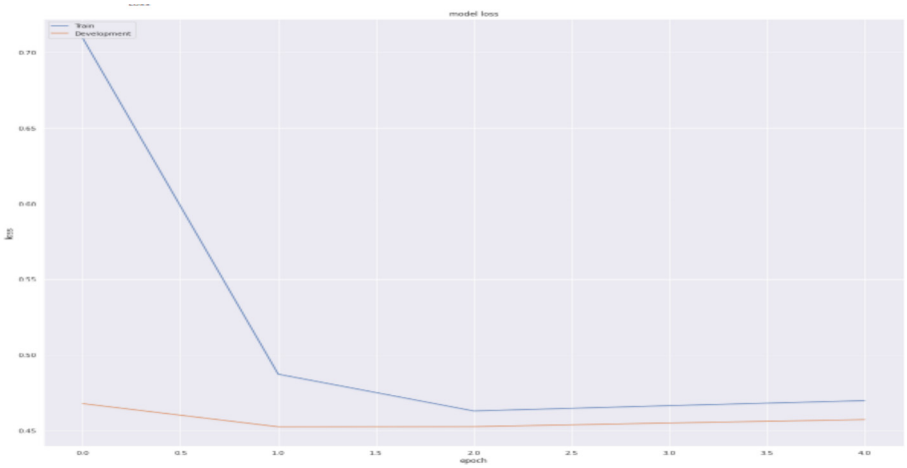


Fig. 6. Fine tuning in multiple epochs – loss value metric

We also apply BERT-base and Distill-BERT without N-gram on the same dataset. The following table shows the results (Table 5).

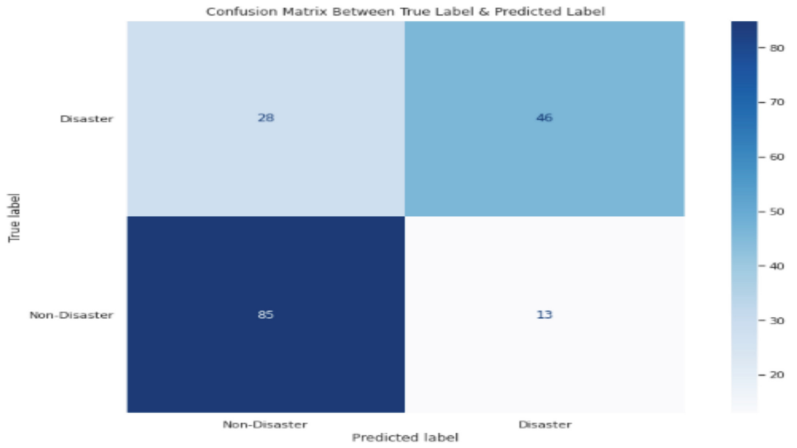


Fig. 7. Confusion Matrix for proposed Method

Table 5. Comparison among BERT-base, Distill-BERT without N-Gram and Distill-BERT with N-Gram

Model	F1-score
BERT-base	0.75
Distill-BERT	0.72
Distill-BERT with N-gram	0.76

4 Conclusion

Recently, transfer learning methods with large pre-trained model has become widely useful especially in NLP applications. As NLP applications requires models to be trained on corpus which are large in size. The NLP capability of the Distill BERT is proven to be better than BERT model by testing it on GLUE benchmark (composed of 9 corpora/tasks). Also test accuracy is only 0.6% point lesser than BERT on the IMDb benchmark while being 40% smaller. In our research it was found that by applying feature extraction techniques like N-gram along with Distill-BERT improvise accuracy as compared to basic BERT model. Thus supersedes the benchmark difference. This makes the Distill-BERT model useful for the NLP applications which requires fast and accurate results like event detection in real time. The proposed work can be extended by applying it on different domains datasets and also more accuracy in results required to be achieved.

References

1. Alam, F., Ofli, F., Imran, M.: Descriptive and visual summaries of disaster events using artificial intelligence techniques: case studies of Hurricanes Harvey, Irma, and Maria. Behav. Inform. Technol. **39**(3), 288–318 (2020).

2. Jiang, S., Li, Y., Qin, T., Meng, Q., Dong, B.: SRCB entity discovery and linking (EDL) and event nugget systems for TAC 2017. In: Proceedings of the 2017 Text Analysis Conference, TAC 2017, Gaithersburg, Maryland, USA, November 13–14, 2017, NIST (2017)
3. Devaraj, J., Ganesan, S., Elavarasan, R.M., Subramaniam, U.: A novel deep learning based model for tropical intensity estimation and post-disaster management of hurricanes. Appl. Sci. **11**(9), 4129 (2021)
4. Adel, H., et al.: Improving crisis events detection using DistilBERT with hunger games search algorithm. Mathematics **10**(3), 447 (2022)
5. ALRashdi, R., O'Keefe, S.: Deep learning and word embeddings for tweet classification for crisis response (2019). arXiv preprint arXiv:1903.11024. Amen, B., Faiz, S., Do, T.-T.: Big data directed acyclic graph model for real-time COVID-19 twitter stream detection. Pattern Recognit. **123**, 108404 (2022)
6. Longformer: the long-document transformer. arXiv:2004.05150
7. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, pp. 1746–1751 (2014). https://doi.org/10.3115/v1/D14-1181
8. Language models are few-shot learners. Adv. Neural Inf. Process. Syst. **33**, 1877–1901. Chennafi, M.E., Bedlaoui, H., Dahou, A., Al-qaness, M.A.: Arabic aspect-based sentiment classification using Seq2Seq dialect normalization and transformers. Knowledge **2**(3), 388–401 (2022)
9. Kodelja, D., Besancon, R., Ferret, O.: Exploiting a more global context for event detection through bootstrapping. In: Azzopardi, L., Stein, B., Fuhr, N., Mayr, P., Hauff, C., Hiemstra, D. (eds.) Advances in Information Retrieval – 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part I, pp. 763–770. Springer (2019)
10. Kodelja, D., Besancon, R., Ferret, O., Borgne, H.L., Boros¸, E.: CEA LIST participation to the TAC 2017 event nugget track. In: 10th Text Analysis Conference (TAC 2017), NIST (2017)
11. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: a lite bert for self-supervised learning of language representations (2019). arXiv:1909.11942
12. Purohit, H., Castillo, C., Pandey, R.: Ranking and grouping social media requests for emergency services using serviceability model. Soc. Netw. Anal. Min. **10**(1), 1–17 (2020)
13. Liu, J., Singhal, T., Blessing, L.T., Wood, K.L., Lim, K.H.: Crisisbert: a robust transformer for crisis classification and contextual crisis embedding. In: Proceedings of the 32nd ACM Conference on Hypertext and Social Media, pp. 133–141 (2021)
14. Basu, M., Shandilya, A., Khosla, P., Ghosh, K., Ghosh, S.: Extracting resource needs and availabilities from microblogs for aiding post-disaster relief operations. IEEE Trans. Comput. Soc. Syst. **6**(3), 604–618 (2019)
15. Makarov, P., Clematide, S.: UZH at TAC KBP 2017: event nugget detection via joint learning with softmax-margin objective. In: Proceedings of the 2017 Text Analysis Conference, TAC 2017, Gaithersburg, Maryland, USA, November 13–14, NIST (2017)
16. Nguyen, T.H., Cho, K., Grishman, R.: Joint event extraction via recurrent neural networks. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 300–309. Association for Computational Linguistics (2016). https://doi.org/10.18653/v1/N16-1034
17. Mitamura, T., Liu, Z., Hovy, E.H.: Events detection, coreference and sequencing: what's next? overview of the TAC KBP 2017 event track. In: Proceedings of the 2017 Text Analysis Conference, TAC 2017, Gaithersburg, Maryland, USA, November 13–14, NIST (2017)
18. De Boom, C., Van Canneyt, S., Dhoedt, B.: Semantics-driven event clustering in twitter feeds (2015)

19. Devgan, S.: 100 Social Media Statistics for 2022 (2021). https://statusbrew.com/insights/social-media-statistics/
20. Gaglio, S., Re, G.L., Morana, M.: Real-time detection of twitter social events from the user's perspective. In: 2015 IEEE International Conference on Communications (ICC). IEEE (2015). https://doi.org/10.1109/icc.2015.7248487, https://doi.org/10.1109/icc.2015.7248487
21. Parikh, R., Karlapalem, K.: Et: Events from tweets. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 613–620. Association for Computing Machinery, New York, NY, USA (2013). https://doi.org/10.1145/2487788.2488006, https://doi.org/10.1145/2487788.2488006
22. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. Proc. AAAI Conf. Artif. Intell. **33**, 7370–7377 (2019)
23. Mangal, D., Makwana, H.: Extracting geo-references from social media text using bi-long short term memory networks. IJEECS (2024). https://doi.org/10.11591/ijeecs.v35.i2.pp1263-1270
24. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.: GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium. Association for Computational Linguistics (2018)
25. Wang, S., Jiang, J.: Machine comprehension using match-lstm and answer pointer (2016). arXiv:1608.07905
26. Wang, W., Yan, M., Wu, C.: Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), volume 1, pp. 1705–1714 (2018)
27. Wang, Z., Ng, P., Ma, X., Nallapati, R., Xiang, B.: Multi-passage Bert: a globally normalized Bert model for open-domain question answering (2019). arXiv:1908.08167
28. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)

Predictive Modeling of Genetic Diseases: A Comparison of Various Machine Learning Approaches

Rimjhim Sinha(✉) and Pradnya V. Kulkarni

School of Computer Science and Technology, MIT World Peace University, Pune, India
9rimjhimsinha@gmail.com, pradnyav.kulkarni@mitwpu.edu.in

Abstract. The aim of this research is to improve the prediction of genetic disorder types by employing large patient data and comparing several machine learning algorithms like XGBoost, Random Forest, AdaBoost, and k-Nearest Neighbors. The study's goal is to enhance our understanding of the underlying patterns and factors contributing to various genetic disorders by analyzing a large dataset. To improve prediction accuracy, the study uses feature selection and dimensionality reduction techniques like Recursive Feature Elimination, Pearson Correlation, and Kendall feature selection. Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE) simplify the data without losing vital details. The primary goal is to develop a robust and interpretable predictive model for genetic disorders. When the XGBoost Classifier was used with Recursive Feature Elimination, the highest accuracy of 82% was achieved. This research aims to determine the best methods for early detection and personalized treatment planning in clinical settings by thoroughly analyzing several machine learning approaches. Enhancing our capacity to anticipate genetic condition types provides healthcare professionals with useful insights, potentially leading to more tailored and successful treatment options for patients. This approach combines modern machine learning techniques with domain-specific expertise to address the complex issue of predicting the type of genetic disorders.

Keywords: Genetic disorders · Machine Learning Classifiers · SMOTE · Feature Selection Techniques · Dimensionality Reduction Techniques

1 Introduction

Genetic disorders arise from abnormalities or mutations in an individual's DNA. They are often referred to as genetic diseases or hereditary disorders. These disorders may arise spontaneously as a result of new mutations, or they may be inherited from one or both parents. Genetic abnormalities can affect physical attributes, organ function, metabolism, and susceptibility to certain illnesses, among other aspects of health. There are numerous genetic diseases and each has distinct symptoms and different levels of severity. Specific diseases can be highly serious or even life-threatening, while others may be rather benign and only cause small difficulties. From birth through adulthood, genetic abnormalities can impact any region of the body and appear at any time in life.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 277–293, 2025.
https://doi.org/10.1007/978-3-031-86296-0_20

This research intends to create and identify the most suitable classification model that can predict the likelihood of genetic disorders and group them into three different categories: single gene inheritance disorders, mitochondrial genetic inheritance diseases, and multifactorial genetic inheritance diseases.

Single gene disorders are caused by genetic abnormalities in a single gene, which can have a variety of implications on how the body functions. Despite their diverse clinical presentations, they have a similar biological foundation that can be handed down to future generations. As a result, accurate diagnosis, risk assessment, and support services are critical for afflicted individuals and their families [7].

Inherited metabolic diseases resulting from mitochondrial gene inheritance are a common disease, affecting roughly 1.6 people in every 5,000. These disorders frequently involve many organ systems, particularly those that require increased energy levels. Symptoms range from moderate to severe, frequently worsening during times of metabolic stress [7].

Multifactorial inheritance combines genetic and environmental factors across medical, congenital, and neuropsychiatric domains. Familial and twin studies highlight the genetic tendency, while environmental factors play a crucial role. Gender influences inheritance patterns, as shown in disorders such as autism spectrum disorder (ASD). Many neuropsychiatric illnesses and birth abnormalities are characterized by complex multifactorial inheritance, which highlights the complicated interplay of genetic and environmental variables [6].

The XGBoost classifier, an ensemble learning algorithm, sequentially builds decision trees, with each successive tree correcting the flaws of the one preceding it. XGBoost, known for its efficiency, excels at handling large datasets and is widely used in both data science challenges and real-world applications [9].

The Random Forest classifier, another ensemble technique, generates a lot of decision trees during training and returns the class mode or mean prediction of each tree. By averaging numerous trees, it improves accuracy and reduces overfitting problems, making it a useful tool for categorization [10].

The AdaBoost Classifier, i.e., an adaptive boosting technique, combines numerous weak classifiers into a strong one with the help of repeated training. Each succeeding classifier corrects previous misclassifications, improving total performance. AdaBoost is highly effective at improving the skills of poorer classifiers [11].

The k-Nearest Neighbors Classifier, a simple and intuitive classification technique, that relies on the idea of data point similarity. A new data point is assigned to the class in the feature space with highest number of nearest neighbors. kNN uses lazy learning to avoid making assumptions about data distribution until classification, resulting in simplicity and efficacy [12].

2 Literature Review

The research by B. Dhanalaxmi et al. [1] introduces the Genetic Disease Analyser which employs various machine learning techniques. Techniques like Principal Component Analysis, Random Forest, Naive Bayes, and Decision Trees were utilized by the authors. The testing of the Genetic Disease Analyser model was carried out on the GEO dataset

which showcased impressive results. This model obtained an accuracy of 98.79% and a sensitivity score of 98.67%. This study showcases the capability of machine learning to improve the comprehension and prediction of genetic diseases. This research serves as a pathway for more effective diagnostic and therapeutic strategies.

In the research by M. Phogat and D. Kumar [2], the authors have identified the key features of genomic data to improve genetic data analysis and thus, the further treatment plan. They have recommended numerous methodologies, tools, and frameworks for feature selection in genomic data. For filter feature selection techniques, they have suggested using chi-square, MFB, t-test, FCBF, ReliefF, gain ratio, mRMR, CFS and CMIM methods. For wrapper feature selection methods, they have recommended Sequential Forward Selection, Sequential Backward Elimination, Support Vector Machine Recursive Feature Elimination, randomized hill climbing, genetic algorithm, and various metaheuristics techniques. They have shown how utilizing proper feature selection techniques can enhance prediction performance and reduce computational cost. The authors have also recommended employing combined feature selection methods to further enhance genomic data analysis.

The authors Raza A. et al. have performed the prediction of genetic disorders by utilizing artificial intelligence-based methods in their research [3]. They have also employed a novel feature engineering approach using extra trees classifier and random forest classifier. The authors have also employed a classifier chain method to tackle multi-label, multi-class challenges. They assessed their model using Macro accuracy, Hamming loss, and α-evaluation score as evaluation metrics. The authors achieved the highest performance when the XGBoost classifier was employed with a 92% α-evaluation score.

In the research by Sameer Quazi [4], the author has studied the application of machine learning algorithms in precision and genomic medicine. The author has emphasized the potential of AI to enhance patient healthcare. A variety of machine learning algorithms were utilized, including decision trees, linear regression, K-nearest neighbors, SVM, genetic algorithms, random forests, hidden Markov models, discriminant analysis, logistic regression, deep learning models, and Naïve Bayes. The author has extracted phenotypic data from Electronic Health Records (EHR) and performed phenotypic classification and analysis. This analysis is useful in diagnosing a broad array of human diseases, describing disease subtypes, discovering biomarkers, drug discovery, and medication response prediction.

In the research by Atta-Ur-Rahman et al. [5], the authors have introduced an advanced model for predicting genome disorders using deep learning. They have addressed the multivariate diseases with high mortality rates. Their model outperforms previous binary class prediction techniques. The authors have utilized the AlexNet convolutional neural network architecture. This model has achieved 89.89% training accuracy and 81.25% testing accuracy. This deep learning model can classify diseases as single gene, mitochondrial gene, and multifactorial gene inheritance disorders. This model can handle large patient datasets efficiently.

In the research by Nasir, Muhammad Umar et al. [7], the authors have presented a robust method with accuracies of 85.7% in training phase, 84.9% in testing phase, and 84.3% in validation phase for fault diagnosis and genetic disease prediction using SVM, kNN and artificial neural networks. With favorable F1 scores and low miss rates,

the precision rates achieved were exceptionally high at 99.7%. The authors attempt to improve diagnostics accuracy by using medical data for genetic disorder prediction using the Scaled Conjugate Gradient Algorithm with a quadratic estimate function. With over 4 million genetic variants and 99.9% genetic similarity between individuals, the authors have highlighted the importance of sophisticated genotyping and testing techniques while also demonstrating the great genetic variety among people.

The reviewed literature highlights the numerous approaches taken to apply machine learning for genetic disease prediction. Although significant progress has been made, there are still a number of gaps that need to be filled in order to enhance the accuracy of prediction models. This research presents a comprehensive research methodology to fill these gaps. This strategy aims to increase the reliability of genetic disease prediction by emphasizing the reduction of class imbalances, using various feature selection techniques, addressing high-dimensional data issues, and carefully evaluating predictive models. This research advances genetic disease prediction techniques using a systematic process that could enhance the capabilities of personalized medicine in diagnosing and treating patients.

3 Purpose, Scope and Objectives

This research is intended to utilize machine learning techniques like XGBoost, Random Forest, AdaBoost, and K-Nearest Neighbors (KNN) to develop an accurate and easily interpretable predictive model for genetic disorders. The main goal is to predict the particular genetic disorder type by utilizing a comprehensive patient dataset. This will help with early diagnosis and personalized treatment planning in clinical settings. The study employs a number of feature selection techniques, such as Recursive Feature Elimination, Pearson Correlation feature selection, and Kendall feature selection, in addition to dimensionality reduction techniques like Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE). The main emphasis of this research is to evaluate and compare these methods' effectiveness for the predictive modeling of genetic disorders.

4 System Architecture

The suggested system architecture as showcased in Fig. 1 offers a comprehensive approach that makes use of machine learning algorithms for predicting the type of genetic disease. Firstly, the data is acquired which covers a wide range of patient demographics, medical history, and clinical observations. Preprocessing this data thoroughly removes any errors or missing values that might reduce the predictability of the results and guarantees that it is clean and appropriate for analysis.

The system uses Synthetic Minority Over-sampling Technique (SMOTE) to remove any biases resulting from class imbalances. SMOTE balances the representation of different genetic disorders and enhances the models' capacity to generalize across all classes by creating synthetic samples for the minority class.

Several machine learning techniques, like XGBoost, Random Forest, AdaBoost, and K-Nearest Neighbors (KNN), were utilized after SMOTE. These models were trained

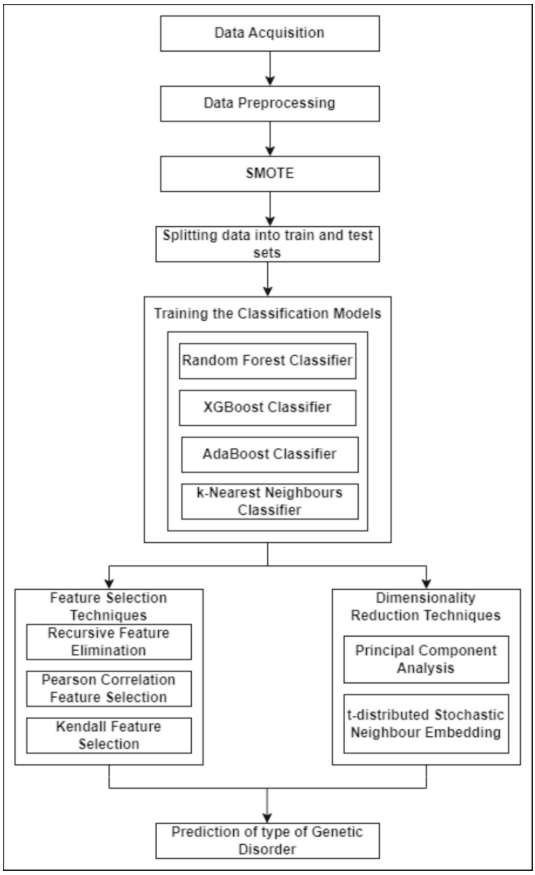


Fig. 1. System Architecture for Predictive Modeling of Genetic Disorders

on SMOTE-adjusted data to find underlying patterns and connections between genetic characteristics and disease categories.

Recursive feature elimination, Pearson Correlation feature selection, and Kendall feature selection are the feature selection methods applied to further enhance the prediction accuracy. By identifying the most important characteristics for accurately predicting genetic diseases, these methods increase the model's accuracy and efficacy.

To abate the dimensionality of the SMOTE-adjusted dataset while maintaining its critical information, t-SNE and PCA are used. By addressing problems with high-dimensional data, this stage increases the model's interpretability and computing performance.

The most efficient model for predicting the types of genetic disorders is then identified by comparing the performances of all the models after all the techniques.

5 About the Dataset

The dataset utilized for the predictive modeling of genetic disorders offers comprehensive information about the patient demographics, medical history, and clinical observations. This dataset consists of 34 feature columns and 1 target column. There are more than 22,000 instances present in this dataset.

The dataset includes demographic information such as age, gender, and parental background in addition to a range of variables related to genetic information about the patient. It also includes markers of medical history such as birth abnormalities, problems, and symptoms. Together with the findings of medical tests, clinical data such as blood cell counts, respiration and heart rates are also provided. Additionally, the collection includes information regarding genetic diseases, including subclassification and kind.

A unique Patient ID uniquely identifies each entry in the dataset. It includes people of all ages, including live and dead patients. The data set provides information on sites of births and medical treatments, as well as insights into inherited genetic traits from both paternal and maternal heritage.

To deal with missing values in the feature columns, first, the dataset was arranged according to the classes that were present in the target field. This segmentation ensured that the imputation technique was customized to fit the particular traits of each class in the target. Then, for every class, the mode of the remaining non-null values in that particular column was substituted for any missing values in the feature columns. By using the mode, the data's statistical features and original distribution were preserved. Furthermore, no duplicates or outliers were present in the dataset.

The numerical columns were scaled using the MinMax Scaler, which preserved their distribution and normalized them between 0 and 1. This ensures that every numerical characteristic is represented proportionately, and bias in the modeling process is avoided. Using LabelEncoder from scikit-learn, Label Encoding was applied to categorical columns. By giving each category a unique integer, this technique converts categorical labels into numerical values that are compatible with machine learning algorithms.

To address the problem of class imbalance, SMOTE was applied [8]. This technique makes sure the model isn't biased towards the majority class by oversampling the minority class, which improves the model's capacity to correctly identify an instance across all classes.

Once preprocessing was completed, 80% of the dataset was allocated for training, while the remaining 20% was reserved for testing the classification models.

6 Model Development

As mentioned earlier, in this research, XGBoost, Random Forest, AdaBoost, and k-Nearest Neighbors classification models were employed for prediction of the type of genetic disorders. Feature selection techniques like Recursive feature elimination, Pearson correlation feature selection, and Kendall feature selection and dimensionality reduction techniques like t-SNE and PCA were utilized.

For this particular research, the XGBoost classifier is set up with 100 estimators (decision trees) and a maximum depth of 10 levels. To ensure accurate outcomes, the

random_state parameter is set to 1. This design, with its depth and number of decision trees, represents a complicated model that captures intricate patterns in the data. Deeper trees can increase the chance of overfitting on the training set, thus it's crucial to strike a balance between interpretability and model complexity. The XGBoost model can be mathematically expressed as follows:

$$\hat{y}_i = \sum_{i=1}^{T} f_t(x_i), f_t \in F \tag{1}$$

In Eq. (1), $\hat{y}_i$ denotes the predicted output for the i^{th} instance of data, T is the total number of decision trees used, $f_t(x_i)$ is the prediction made by the t^{th} decision tree for the input features x_i and F shows a space of possible functions which in this case are decision trees [20].

The RandomForest Classifier is configured with parameters for ensemble learning. In order to maximize information gain for classification, it uses 100 decision trees (estimators), and the criterion parameter is set to entropy. Entropy, used to determine split quality, can be mathematically represented as:

$$H(s_1, s_2, \ldots, s_n) = -\sum_{k=1}^{c} p_k * \log_2(p_k) \tag{2}$$

In Eq. (2), H represents the entropy, s_1, s_2, …, s_n are the count of samples in each class, c is the total number of classes, p_k is the probability of a sample occurring in class k and it is calculated as

$$p_k = \frac{s_k}{s} \tag{3}$$

Here, in Eq. (3), s_k is the total number of samples in class k and s is the sum of all the samples present in all classes [19]. To manage complexity and lessen overfitting, the depth of each tree is limited to ten levels. Reproducible results are guaranteed since the random_state parameter is set to 1. Overall, while resolving overfitting issues, RandomForest Classifier achieves strong classification performance by utilizing the power of ensemble learning and randomization.

AdaBoost Classifier trains a series of weak learners, each of which is a decision tree with a maximum depth of 3. It iteratively modifies misclassified sample weights through 50 weak learners (n_estimators) to emphasize their relevance in later rounds, controlling the effect of each learner at a learning rate of 0.1. Consistent results are guaranteed by the random_state parameter, and 'SAMME' is chosen as the boosting technique. Essentially, AdaBoost Classifier uses adaptive boosting techniques and ensemble learning to build a dependable classification model that can handle complex datasets with few mistakes and overfitting. The mathematical expression of the AdaBoost Classifier is:

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \tag{4}$$

In the above Eq. (4), H(x) showcases the final prediction made by the AdaBoost Classifier for the input x. The sign function returns +1 if the calculated value is zero or

positive and −1 if that value is negative. The weight given to the t^{th} weak learner is α_t and for the input x, the output of the t^{th} weak learner is $h_t(x)$ [21].

As part of its KNN configuration, the KNeighbors Classifier uses five neighbors to do classification. The labels of a data point's five closest neighbors in the feature space are evaluated to forecast the class of the data point using this approach. KNeighbors Classifier predicts the label for a given data point by using distance measures, often Euclidean distance, and identifying the most frequent class label among its neighbors. For the given two vectors $p = (p_1, p_2, \ldots, p_n)$ and $q = (q_1, q_2, \ldots, q_n)$ the formula for calculating the Euclidean distance between then is shown below in Eq. (5).

$$d(p, q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{5}$$

In Eq. (5), the square root of the sum of the squared difference between the i^{th} elements of vectors p and q is considered for calculating the Euclidean distance [22]. For classification, KNN offers a simple and effective method that works especially well with smaller datasets that have non-linear decision borders. However, because it relies on distance computations, its processing needs could be problematic when working with bigger datasets.

To choose the dataset's most significant features, the Recursive Feature Elimination (RFE) method is applied by utilizing scikit-learn's RFECV package. It works by repeatedly eliminating features and evaluating their effect on the model's performance, using cross-validation. For this particular research, the base estimator for RFE is a Decision Tree Classifier, which assesses feature importance based on their efficacy in splitting the data within a decision tree framework [13]. RFECV simplifies the feature selection procedure by allowing the specification of parameters such as the number of cross-validation folds, the step size for feature elimination, and the scoring metric which is defined as 10, 5, and accuracy respectively in this research. The scoring metric can be mathematically represented as:

$$Score = \frac{1}{k} \sum_{i=1}^{k} Accuracy_i \tag{6}$$

In Eq. (6), Score represents the scoring metric, k denotes the number of cross-validation folds and $Accuracy_i$ is accuracy of the model on the i^{th} cross-validation fold. Recursive Feature Elimination makes it easier to create machine learning models that are more efficient and comprehensible.

Pearson correlation feature selection takes into account the correlation coefficient between each feature and the target variable, which represents the strength of their linear relationship. Pearson correlation coefficient is calculated as follows:

$$r_p = \frac{\sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n} (x_i - x)^2 \sum_{i=1}^{n} (yi - y)^2}} \tag{7}$$

Here, in Eq. (7), r_p is the Pearson correlation coefficient indicating the relationship between X and Y, n is the number of data points, x and y are the standard deviations and $\overline{X}$ and $\overline{Y}$ are the means of X and Y respectively [23]. Equation (7) can be simplified as showcased in Eq. (8).

$$r_p = \frac{cov(X, Y)}{xy} \tag{8}$$

In Eq. (8), cov(X, Y) is the covariance between variables X and Y. Features with greater absolute correlation coefficients are considered more impactful for making predictions, but those closer to zero offer less information. While this method helps to identify essential features for modeling, it assumes linear correlations and is sensitive to outliers and non-normal data distributions [14].

Kendall correlation feature selection employs the Kendall correlation coefficient to determine the ordinal connection between the target variable and the features, indicating the strength of their rank-order relationship. The mathematical representation of the Kendall correlation coefficient is shown below.

$$\tau_{tau} = \frac{2(C - D)}{n(n - 1)} \tag{9}$$

In Eq. (9), τ_{tau} is the Kendall correlation coefficient, D is the number of discordant pairs, C is the number of concordant pairs and n is the total number of instances in the dataset [23]. Features with greater absolute Kendall correlation coefficients are thought to be more relevant for making predictions, while those closer to zero are less significant. This method helps in identifying the important features for modeling by assuming ordinal connections. However, it can be influenced by outliers and non-normal data distributions [15].

This research also utilizes PCA from the scikit-learn decomposition module to decrease data dimensionality to two principal components. These components retain the major variance in the original dataset while maintaining its core characteristics. PCA converts the data and reduces dimensionality through singular value decomposition which is represented in the below formula

$$X = UDV^T \tag{10}$$

In Eq. (10), X represents the original data matrix, U is the factor scores in reduced space, D denotes the variance values and V^T shows feature contributions [24]. The converted data provides compact representations of the original features, facilitating the visualization of complex, high-dimensional data in a reduced manner. This strategy makes it easier to comprehend and has the ability to uncover underlying patterns or clusters in the dataset [16].

In this research, the t-SNE method from scikit-learn's manifold module is used to reduce data dimensionality to three components. The objective is to capture the data's underlying structure by mapping high-dimensional data points to lower-dimensional space. This transformation is achieved by minimizing the Kullback-Leibler divergence

between the pairwise similarities of high-dimensional data, P, and those of its lower-dimensional representation, Q. It can be mathematically represented as:

$$C = KL(P||Q) = \Sigma_i \Sigma_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{11}$$

In Eq. (11), p_{ij} represents the conditional probability of point j selection i as its neighbor in the high-dimensional space, while q_{ij} is the equivalent likelihood in the lower-dimensional space [25]. The resultant converted data provides a more condensed representation of the original dataset. This strategy is useful for reducing complex information into more understandable formats, which might reveal complicated patterns or correlations between data points [17].

7 Results and Analysis

In order to predict genetic disorders, this research thoroughly evaluated a number of classification models, like XGBoost, Random Forest, AdaBoost, and kNN. Preprocessing approaches, like SMOTE for managing class imbalance and numerous feature selection techniques, including Recursive Feature Elimination, Pearson Correlation feature selection, and Kendall feature selection were explored. Dimensionality reduction techniques like t-SNE and PCA were also used. A random state of 1 was shown to be best once the dataset was divided into training and testing sets, guaranteeing consistent results throughout the studies. The evaluation metrics used to assess the performance of all the classification models were F1 Score, Precision, Recall, and Accuracy.

F1 score is the harmonic mean of the recall and precision metrics. It can be mathematically represented as shown in Eq. (12) [18].

$$F1Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{12}$$

Precision is the ratio of true positive predictions to the total number of positive predictions made by the classification model. Equation (13) showcases the formula for calculating precision [18].

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

Recall is the ratio of true positive predictions to the total number of actual positive instances in the data. It is represented in Eq. (14) [18].

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

Accuracy is the proportion of accurately predicted instances by the classification model among all the instances present in the dataset. The mathematical formula for accuracy is given in Eq. (15) [18].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

In Eqs. (13), (14), (15) TP denotes the number of true positive predictions, TN indicates the number of true negative predictions, FP represents the number of false positive predictions, and FN signifies the number of false negative predictions.

The key findings from all the analyses and the performance of all the classification models are outlined below.

7.1 Performance After SMOTE

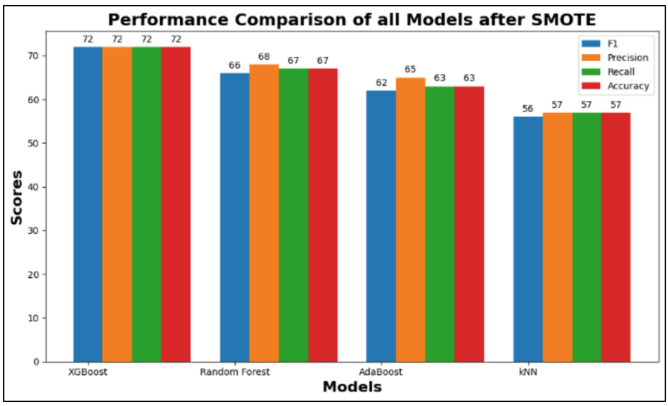


Fig. 2. Performance after SMOTE

Figure 2 showcases the results of all the classification models when evaluated on various metrics like F1 Score, Precision, Recall, and Accuracy after SMOTE was applied to the dataset. The accuracies achieved by all the classification models are as follows:

- XGBoost Classifier – 72%
- Random Forest – 67%
- AdaBoost Classifier – 63%
- k – Nearest Neighbors Classifier – 57%

It can be noted that the XGBoost Classifier had the highest accuracy with 72% and the lowest accuracy was showcased by the KNN classifier with 57% accuracy.

7.2 Performance After SMOTE and Recursive Feature Elimination

Figure 3 showcases the results of all the classification models after SMOTE and Recursive Feature Elimination were applied to the dataset. The accuracies achieved by all the classification models are as follows:

- XGBoost Classifier – 82%
- Random Forest – 75%
- AdaBoost Classifier – 68%
- k – Nearest Neighbors Classifier – 70%

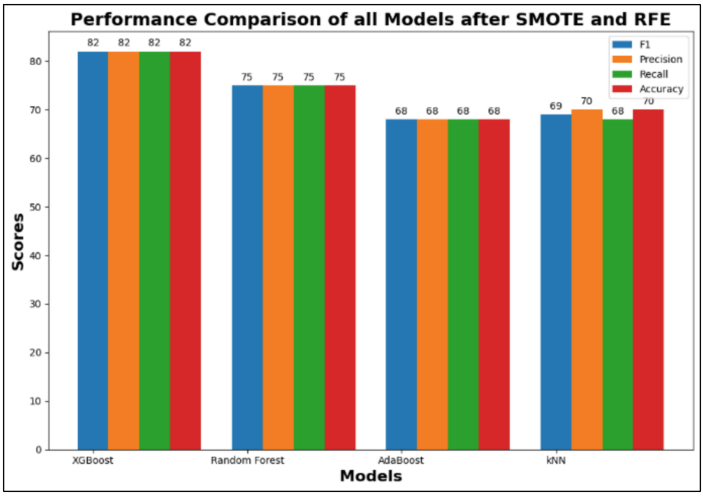


Fig. 3. Performance after SMOTE and Recursive Feature Elimination

It can be noted that the XGBoost Classifier had the highest accuracy with 82% and the lowest accuracy was showcased by the AdaBoost Classifier with 68% accuracy.

7.3 Performance After SMOTE and Pearson Correlation Feature Selection

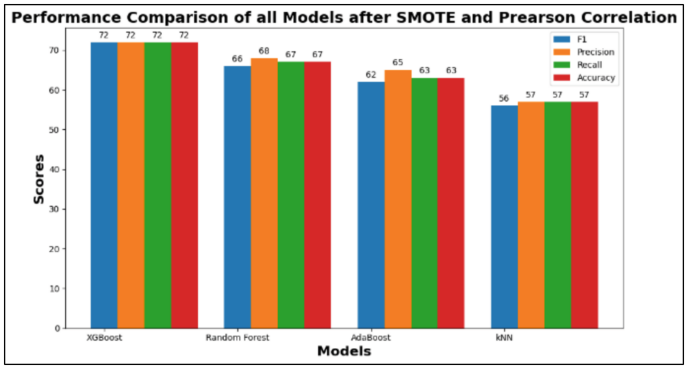


Fig. 4. Performance after SMOTE and Pearson correlation feature selection

Figure 4 showcases the results of all the classification models after SMOTE and Pearson Correlation feature selection were applied to the dataset. The accuracies achieved by all the classification models are as follows:

- XGBoost Classifier – 72%
- Random Forest – 67%
- AdaBoost Classifier – 63%

– k – Nearest Neighbors Classifier – 57%

It can be noted that the XGBoost Classifier had the highest accuracy with 72% and the lowest accuracy was showcased by the KNN classifier with 57% accuracy.

7.4 Performance After SMOTE and Kendall Feature Selection

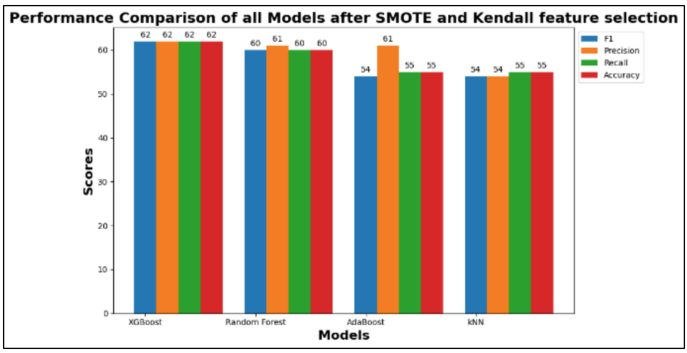


Fig. 5. Performance after SMOTE and Kendall feature selection

Figure 5 showcases the results of all the classification models after SMOTE and Kendall feature selection was applied to the dataset. The accuracies achieved by all the classification models are as follows:

– XGBoost Classifier – 62%
– Random Forest – 60%
– AdaBoost Classifier – 55%
– k – Nearest Neighbors Classifier – 54%

It can be noted that the XGBoost Classifier had the highest accuracy with 62% and the lowest accuracy was showcased by the KNN classifier with 54% accuracy.

7.5 Performance After SMOTE and PCA Dimensionality Reduction

Figure 6 showcases the results of all the classification models after SMOTE and PCA dimensionality reduction were applied to the dataset. The accuracies achieved by all the classification models are as follows:

– XGBoost Classifier – 58%
– Random Forest – 58%
– AdaBoost Classifier – 44%
– k – Nearest Neighbors Classifier – 55%

It can be noted that the XGBoost Classifier had the highest accuracy with 58% and the lowest accuracy was showcased by the AdaBoost classifier with 44% accuracy.

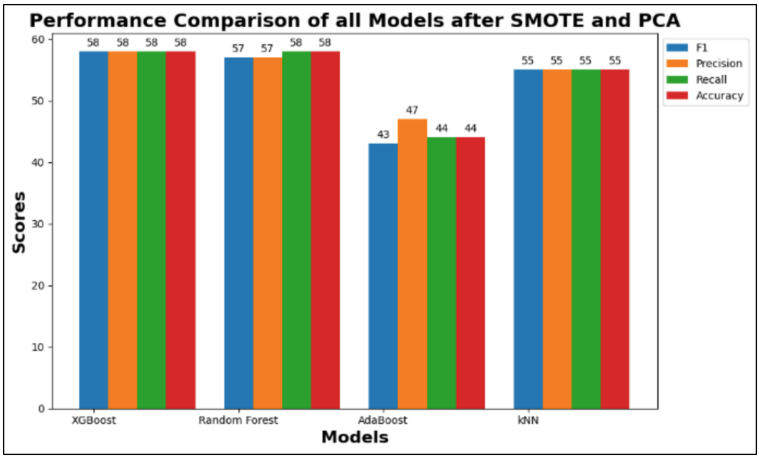


Fig. 6. Performance after SMOTE and PCA dimensionality reduction

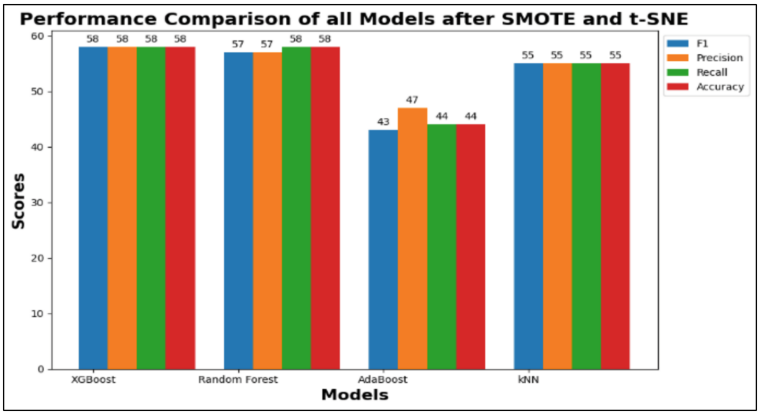


Fig. 7. Performance after SMOTE and t-SNE dimensionality reduction

7.6 Performance After SMOTE and t-SNE Dimensionality Reduction

Figure 7 showcases the results of all the classification models after SMOTE and t-SNE dimensionality reduction were applied to the dataset. The accuracies achieved by all the classification models are as follows:

- XGBoost Classifier – 58%
- Random Forest – 58%
- AdaBoost Classifier – 44%
- k – Nearest Neighbors Classifier – 55%

It can be noted that the XGBoost Classifier had the highest accuracy with 58% and the lowest accuracy was showcased by the AdaBoost classifier with 44% accuracy.

7.7 Analysis of the Obtained Results

Table 1. Performance Metrics of Classifiers Across Various Methods

	XGBoost Classifier				Random Forest Classifier				AdaBoost Classifier				kNN Classifier			
	F1	*P*	*R*	*A*	*F1*	*P*	*R*	*A*	*F1*	*P*	*R*	*A*	*F1*	*P*	*R*	*A*
SMOTE	72%	72%	72%	72%	66%	68%	67%	67%	62%	65%	63%	63%	56%	57%	57%	57%
RFE	82%	82%	82%	82%	75%	75%	75%	75%	68%	68%	68%	68%	69%	70%	68%	70%
Pearson Correlation	72%	72%	72%	72%	66%	68%	67%	67%	62%	65%	63%	63%	56%	57%	57%	57%
Kendall	62%	62%	62%	62%	60%	61%	60%	60%	54%	61%	55%	55%	54%	54%	55%	54%
PCA	58%	58%	58%	58%	57%	57%	58%	58%	43%	47%	44%	44%	55%	55%	55%	55%
t-SNE	58%	58%	58%	58%	57%	57%	58%	58%	43%	47%	44%	44%	55%	55%	55%	55%

In Table 1, the performance of all the classification models is compared by utilizing the evaluation metrics F1 Score, Precision, Recall and Accuracy denoted as F1, P, R, A respectively. It was observed that the XGBoost Classifier has outperformed all the other classification models consistently. Dimensionality reduction techniques have negatively impacted and drastic changes can be seen in the performance of the classification models. Hence, it can be inferred that dimensionality reduction techniques are not suitable for datasets containing genetic information. When XGBoost Classifier was used along with Recursive Feature Elimination, the highest accuracy of 82% was achieved.

8 Conclusion and Future Scope

In conclusion, to predict genetic disorders, this research thoroughly examined feature selection techniques, dimensionality reduction techniques, preprocessing approaches, and classification models. It turned out that XGBoost was the best classifier, consistently outperforming all other classifications in different scenarios. Recursive Feature Elimination integration was shown to greatly improve classification accuracy, F1 score, precision, and recall for all the classification models. Nevertheless, the procedures of Kendall Feature Selection and Pearson Correlation failed to provide significant enhancements, suggesting their restricted suitability for this particular dataset. Furthermore, all the classifier's performance was negatively impacted by PCA and t-SNE Dimensionality Reduction techniques, indicating that it is not a suitable method for maintaining discriminative information. Overall, the results highlight how crucial it is to choose feature selection and preprocessing strategies carefully in order to increase classification accuracy and model interpretability.

In the future, the performance of the model might be further improved by exploring advanced feature selection strategies or ensemble methods that integrate multiple techniques. Further ways to reduce computational complexity and preserve important information include optimizing current methods or exploring alternate dimensionality reduction approaches. Enhancing the predictive power of genetic disorders might be achieved by incorporating domain-specific information or by integrating new datasets, including genomic or clinical data. Lastly, a more thorough evaluation of the performance

and dependability of the model may be provided by using advanced assessment metrics or by doing cross-validation tests to verify model generalization. In order to support personalized medicine and clinical decision-making, these aspects might be addressed to improve the efficiency and reliability of the predictive modeling of genetic disorders.

References

1. Dhanalaxmi, B., Anirudh, K., Nikhitha, G., Jyothi, R.: A survey on analysis of genetic diseases using machine learning techniques. In: 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, pp. 496–501 (2021). https://doi.org/10.1109/I-SMAC52330.2021.9640723
2. Phogat, M., Kumar, D.: Feature selection techniques for genomic data. In: 2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON), Faridabad, India, pp. 785–790 (2022). https://doi.org/10.1109/COM-IT-CON54601.2022.9850466
3. Raza, A., et al.: Predicting genetic disorder and types of disorder using chain classifier approach. Genes (Basel) **14**(1), 71 (2022). https://doi.org/10.3390/genes14010071.PMID:36672812;PMCID:PMC9858679
4. Quazi, S.: Artificial intelligence and machine learning in precision and genomic medicine. Med. Oncol. **39**(8), 120 (2022)
5. Atta-Ur-Rahman, et al.: Advance genome disorder prediction model empowered with deep learning. IEEE Access **10**, 70317–70328 (2022). https://doi.org/10.1109/ACCESS.2022.3186998
6. Klugman, S., Rabin-Havt, S.S.: Chapter 2 - Non-mendelian genetics. In: Norton, M.E.. Kuller, J.A., Dugoff, L. (eds.) Perinatal Genetics, pp. 11–19. Elsevier (2019). ISBN 9780323530941. https://doi.org/10.1016/B978-0-323-53094-1.00002-3. https://www.sciencedirect.com/science/article/pii/B9780323530941000023
7. Nasir, M.U., Khan, M., Zubair, M., Ghazal, T., Said, R., Al Hamadi, H.: Single and mitochondrial gene inheritance disorder prediction using machine learning. Comput. Mater. Contin. **73**, 953–963 (2022). https://doi.org/10.32604/cmc.2022.028958
8. Abdoh, S.F., Abo Rizka, M., Maghraby, F.A.: Cervical cancer diagnosis using random forest classifier With SMOTE and feature reduction techniques. IEEE Access **6**, 59475–59485 (2018). https://doi.org/10.1109/ACCESS.2018.2874063
9. Laxmikant, K., Bhuvaneswari, R., Natarajan, B.: An efficient approach to detect diabetes using XGBoost classifier. In: 2023 Winter Summit on Smart Computing and Networks (WiSSCoN), Chennai, India, pp. 1–8 (2023). https://doi.org/10.1109/WiSSCoN56857.2023.10133854
10. Parmar, A., Katariya, R., Patel, V.: A review on random forest: an ensemble classifier. In: Hemanth, J., Fernando, X., Lafata, P., Baig, Z. (eds.) International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018, ICICI 2018. LNDECT, vol. 26, pp. 758–763. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-03146-6_86
11. An, T.-K., Kim, M.-H.: A new diverse AdaBoost classifier. In: 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China, pp. 359–363 (2010). https://doi.org/10.1109/AICI.2010.82
12. Zhang, Z.: Introduction to machine learning: K-nearest neighbors. Ann. Transl. Med. **4**(11), 218 (2016). https://doi.org/10.21037/atm.2016.03.37. PMID: 27386492; PMCID: PMC4916348
13. Awad, M., Fraihat, S.: Recursive feature elimination with cross-validation with decision tree: feature selection method for machine learning-based intrusion detection systems. J. Sens. Actuator Netw. **12**, 67 (2023). https://doi.org/10.3390/jsan12050067

14. Chen, P., et al.: Research on intrusion detection method based on Pearson correlation coefficient feature selection algorithm. J. Phys. Conf. Ser. **1757**, 012054 (2021). https://doi.org/10.1088/1742-6596/1757/1/012054
15. Van Hulse, J., Khoshgoftaar, T.M., Napolitano, A., et al.: Threshold-based feature selection techniques for high-dimensional bioinformatics data. Netw. Model. Anal. Health Inform. Bioinforma. **1**, 47–61 (2012). https://doi.org/10.1007/s13721-012-0006-6
16. Sorzano, C.O.S., Vargas, J., Pascual Montano, A.: A survey of dimensionality reduction techniques. arXiv preprint arXiv:1403.2877 (2014)
17. Xu, X., Xie, Z., Yang, Z., Li, D., Xu, X.: A t-SNE based classification approach to compositional microbiome data. Front. Genet. **11** (2020). https://doi.org/10.3389/fgene.2020.620143
18. Kulkarni, P., Rai, S., Sachdeo, R., Kale, R.: Deep learning-based educational user profile and user rating recommendation system for e-learning. J. Inf. Syst. Telecommun. (JIST) **11**, 185–195 (2023). https://doi.org/10.61186/jist.27448.11.43.185
19. Huang, N., et al.: Feature selection of power quality disturbance signals with an entropy-importance-based random forest. Entropy **18**, 44 (2016). https://doi.org/10.3390/e18020044
20. Wang, L., Wang, X., Chen, A., Jin, X., Che, H.: Prediction of type 2 diabetes risk and its effect evaluation based on the XGBoost model. Healthcare **8**, 247 (2020). https://doi.org/10.3390/healthcare8030247
21. Rohan, T.I., Awan-Ur-Rahman, Siddik, A.B., Islam, M., Yusuf, M.S.U.: A precise breast cancer detection approach using ensemble of random forest with AdaBoost. In: 2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2), Rajshahi, Bangladesh, pp. 1–4 (2019). https://doi.org/10.1109/IC4ME247184.2019.9036697
22. Dewia, A.M.S.I., Dwidasmaraa, I.B.G.: Implementation of the K-nearest neighbor (KNN) algorithm for classification of obesity levels. Jurnal Elektronik Ilmu Komputer Udayana p-ISSN **2301**, 5373 (2020)
23. El-Hashash, E., Hassan, R.: A comparison of the pearson, spearman rank and kendall tau correlation coefficients using quantitative variables. Asian J. Probab. Stat. **20**, 36–48 (2022). https://doi.org/10.9734/AJPAS/2022/v20i3425
24. Stewart, S., Ivy, M.A., Anslyn, E.V.: The use of principal component analysis and discriminant analysis in differential sensing routines. Chem. Soc. Rev. **43**(1), 70–84 (2014)
25. Liu, L.-p., et al.: Application of K-means++ algorithm based on t-SNE dimension reduction in transformer district clustering. In: 2020 Asia Energy and Electrical Engineering Symposium (AEEES), Chengdu, China, pp. 74–78 (2020). https://doi.org/10.1109/AEEES48850.2020.9121438

Extracting the Pneumonia Signature: A Deep Learning Framework for Definitive Diagnosis

Malay Vyas(✉) and Apurva. A. Mehta

Dharmsinh Desai University, Computer Engineering, Nadiad 387001, India
malayvyas2004@gmail.com, apurvamehta.ce@ddu.ac.in

Abstract. Pneumonia is a significant worldwide health and wellness concern that creates considerable health issues plus fatality, highlighting the importance of promptly and accurately detecting and treating it. Despite the improvements in imaging innovation, the hand-operated evaluation of chest radiographs by radiologists remains the fundamental technique for spotting pneumonia, bringing about hold-ups in medical diagnosis together with therapy. This research suggests a pneumonia discovery technique that utilizes deep learning strategies to automate the procedure. By harnessing an extensive data source of classified chest radiographs the recommended design intends to precisely determine locations influenced by pneumonia, making it possible for better and more efficient medical diagnosis and therapy. The design uses a custom convolutional neural network (CNN) that undergoes training on various pneumonia- positive and pneumonia-negative instances from multiple healthcare organizations. Before educating the design, various pre-processing actions were taken for the chest radiographs to boost integrity and efficiency. This research study adds to the growth of an automated, coupled with a trusted, pneumonia discovery system, which has the potential to enhance individual results and boost healthcare effectiveness.

Keywords: Convolutional Neural Network · Transfer Learning · Pneumonia detection · Chest X-ray images

1 Introduction

Pneumonia is one of the primary reasons for mortality in children worldwide. In India, it leads to deaths in nearly 20% of those cases [1]. The world has made significant progress toward reducing it, but childhood pneumonia continues [2]. Early diagnosis is one of the primary ways of limiting the fatality rate. Pneumonia infection targets one or both lungs by infecting them. This can happen due to bacterial, viral, or fungal infections [3].

Inspection of the lungs, heart, and blood vessels is necessary for determining the possibility of pneumonia infection. This is made possible through a chest X-ray. However, many trained eyes of radiologists find it difficult to detect pneumonia from chest X-rays due to the X-ray color scheme. So, pneumonia detection through chest X-ray is a pending concern waiting to be resolved [4]. The technology-supported algorithms can aid human experts in reaching the diagnosis of pneumonia due to its reproducible decision support system and abundance of power.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 294–304, 2025.
https://doi.org/10.1007/978-3-031-86296-0_21

An accurate diagnosis can be performed by precise image scanning and interpretation. Conventional machine learning algorithms can be used for the same [5]. Traditional machine learning algorithms like Naive Bayes, Support Vector machines, and Random Forest run into obstacles in pneumonia discovery from X-ray scans fumbling with complex function design restricted capabilities for recording intricate spatial connections and vulnerability to heavy information concerns [6]. Their adaptability to progressive pneumonia situations as well as irregular datasets hinders their efficacy. On the other hand, CNNs can automatically learn hierarchical features and understand spatial relationships in images which has been proven to be highly effective in tasks like pneumonia detection using X-ray scans. Their capacity to handle raw pixel values, adapt to varied image sizes, and capture complex patterns makes them well-suited for medical image analysis.

The last 10 years have seen abundant growth of deep learning in almost all re- search areas. Research in material science, information security, natural language processing, seismology, computational biology [7], medical image analysis, etc. has seen tremendous progress.

The version of the Convolutional Neural Network (CNN) presented in this paper for pneumonia discovery in X-ray scans uses semantic layers for automated function removal. Educated on classified datasets, it is properly pneumonia-positive and adverse, using metrics like precision, accuracy, and recall. With a concentration on enhancing hyperparameters together with leveraging great adjusting techniques our CNN functions as a trustworthy device coupled with effective pneumonia medical diagnosis based on the radiographic pictures.

The rest of the paper is organized as follows: Sect. 2 delves deep into recent re- search on pneumonia detection for recent insights. Section 3 provides a detailed presentation of the proposed approach. Sections 4 and 5 highlight the assessment of the proposed solution and discuss the reported results. Section 6 concludes with this work's findings.

2 Literature Survey

The literature survey of pneumonia detection research paper delves into a comprehensive exploration of existing methodologies, advancements, and challenges within the domain. The work of Saul [8] uses a two-step approach for image processing before modeling. This provides better image features for training. The classification is carried out using a Convolutional Neural Network followed by a residual neural network. The work is based on 4000 X-ray images. This work reports the highest accuracy of 78.73% when performing modeling after applying preprocessing techniques increment in contrast and lightening methods. Weighted classifiers are used in an approach to classify pneumonia from digital images of chest X-rays [9]. Weighted predictions are formed using the deep learning architectures InceptionV3, ResNet18, DenseNet121, Xception, and MobileNetV3. The data set bias is tackled using data augmentation techniques. Their model reports an F1 score of 99.002 and an AUC score of 99.809. An ensemble approach [10] reports that Efficientnet-B4-based U-Net architecture performs well compared to the ResNet-based U-Net architecture for pneumonia detection. The model faces challenges in handling type-1 errors. Pneumonia can be caused by viral or bacterial infection. In a study by Rahman [11], three types of classifications are studied

using different pre-trained Convolutional Neural Networks. Transfer learning is utilized for better classification on a comparably smaller dataset. The three classification categories are normal and pneumonia, normal, bacterial, and viral pneumonia, and viral and bacterial pneumonia. DenseNet201outperfroms other pre-trained models with an accuracy of 98%, 93.3%, and 95% for 3 classification schemes respectively. A mobile application has been developed supporting pneumonia detection [12]. CreateML is used for modeling. It can create, train, and implement models from Mac. The work re- ports an accuracy of 85%.

The dataset images of 100 X 100 are enriched for traces using the marker- controlled watershed (MCW) segmentation algorithm [13]. These images of chest X-rays are used for modeling. The modeling is performed using attention and LSTM models. The proposed approach classifies an input image into one of three classes COVID-19, normal, and pneumonia. RSNA Pneumonia Detection Challenge dataset is put to work for pneumonia identification. This dataset consists of 32227 images [14]. The possibility of overfitting is handled by adding additional noise to the dataset. Also, a variety of augmentation methods are applied. VGG19, VGG16, ResNet50, InceptionNet v3, and YOLO v5 CNN models are used for machine learning. VGG16 reports the highest accuracy of 88%.

Authors across the globe are making progress in automating pneumonia, COVID-19, and tuberculosis detection using CT scans and X-ray images through deep- learning architectures [15].

3 Methods and Materials

3.1 Dataset

The dataset used in the paper was obtained from Mendeley Data [16]. It contains a total of 5863 images. The dataset contains 3 sub-directories named train, test, and val and each contains 2 subfolders named Normal and Pneumonia, where the subclasses contain X-ray scans of multiple users. The training data has 5216 total images, further classified as 1341 Normal and 3875 Pneumonia patient's chest X-rays. Among the total 624 testing class images, 234 were under the Normal class and the rest 390 were under the Pneumonia class. In the validation sub- directory, there are 16 images amongst which 8 belonged to the Normal Class and the rest of 8 were under the Pneumonia class. The dataset displays a bias towards the Pneumonia class, necessitating an approach to solve the imbalance. A possible solution is to hedge the bias by adding additional image scans to the Normal class using the Data Augmentation Technique [17].

3.2 Preprocessing

Image Augmentation is a technique that receives an image as input and performs some changes that transform the original image to an augmented image where some properties of the image are changed but the crux of the image remains untouched [18, 19].

After preprocessing (i.e. image augmentation) as shown in Fig. 1, the training subdirectory had 3875 images in each sub-folder. The testing subdirectories had 390 images in each sub-folder. The validation data is 20 percent of the training data, chosen by the random selection process.

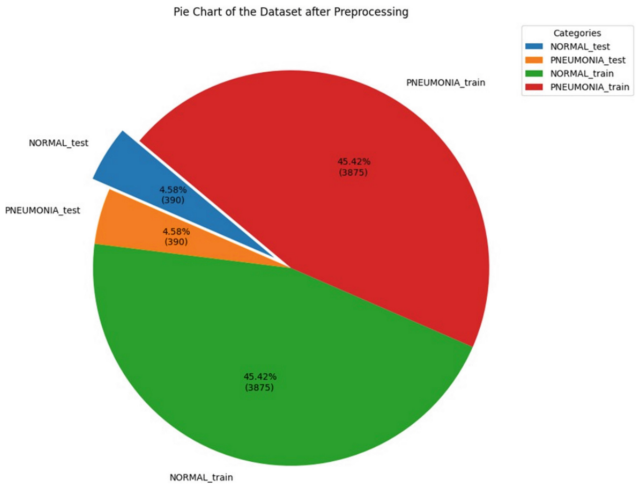


Fig. 1. The pie chart delineates the balanced distribution subsequent to preprocessing but preceding validation split.

3.3 CNN Architecture

The Convolutional Neural Network (CNN) architecture has emerged as a corner- stone in deep learning, revolutionizing tasks such as image recognition, natural language processing, and medical diagnosis.

CNN is considered best for image classification for reasons such as 1. Dense Connectivity 2. Parameter Sharing 3. Feature Reuse 4. Enhanced Information flow 5. Adaptability to different image sizes and many more. In CNN, the filters and layers extract the features from the images. Initial layers detect small features and gradually move forward to higher layers, detecting larger features and pat- terns [20]. The model has 11 million parameters signifying its complexity and capacity for capturing sinuous details in image data.

The color-coded 3D representation of the customized CNN model presented in Fig. 2 provides details of the convolutional block, max pool layer, dropout layer, flattening layer, and dense layer. The first Convolutional layer or input layer gets an input of (256,256,3) image and employs 32 filters with a kernel size of (3,3) and the activation function Rectified Linear Unit or ReLU. The 2nd Convolutional block processes input from the previous layer reducing dimensions to 127x127 with 64 filters with a (3,3) kernel and the activation function being ReLU, followed by a Max Pool Layer with a pooling size of (2,2) and a dropout layer with a dropout rate of 0.2.

The 3rd, 4th, and 5th Convolutional bocks gracefully traverse the dimensions of 62×62, 30×30, and 14×14 input dimensions respectively. Each block shows its unique characteristics with 128, 256, and 512 filters and activating the ReLU function. Adhering to this pattern Max Pooling(2,2) as well as a Dropout layer with a dropout rate of 0.2 are presented after each Convolutional block, amounting to the class of the network style. A flattening layer has been added for flattening out the output from the 5th Convolutional

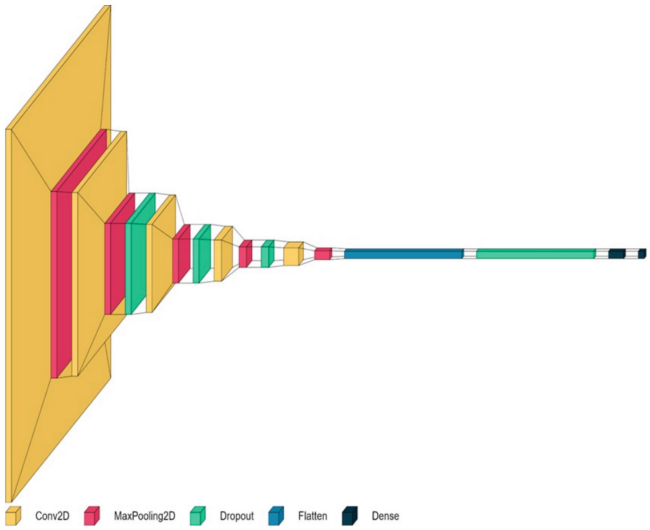


Fig. 2. 3D representation of the Convolutional Neural Network and the hidden layers of the custom model.

block, followed by a dropout layer with a dropout rate of 0.2. At last, a dense layer is added with only one unit and Sigmoid as the activation function.

3.4 The Metrics

The metrics used for the validation of the model are as equally important as the other hyperparameters. The metrics display the perfection achieved by the model. There are numerous metrics for evaluating the CNN models, amongst which Accuracy, Precision, Sensitivity, FPR, and Recall are some of the most reliable metrics. Accuracy dictates the overall correctness of the model and whether an instance is classified correctly or not. The TP and TN stand for True Positive and True Negative whereas FP and FN stand for False Positive and False Negative. If an instance belongs to the True state and the model correctly classifies it as a positive instance then the instance is added to the True Positive(TP) class. Similarly, if a model correctly classifies a negative instance, it is added to the True Negative(TN) class. The accuracy is calculated as the ratio of the total correctly classified over the total number of instances in the dataset as shown in Eq. 1. If the model misclassifies an actual positive sample as a false one, it is counted as a False Negative(FN) sample. Similarly, if a model misclassifies an actual false example as a true one, it is added to the False Positive (FP) class. The precision can be defined as the ratio of true positive predicted instances over the total number of positive predictions as shown in Equation 2.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FNTP} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$AUC_ROC = \int_0^1 TPR(FPR)DFPR \tag{4}$$

The Recall or true positive rate for a model is the ratio of true positively predicted instances and the total number of actual positive instances. It is represented mathematically as in Equation 3. The sensible factor to consider the metrics stands as an essential element in reviewing the efficiency of artificial intelligence designs.

4 Training and Output

The model with the layers explained above was compiled with the optimizer as Adam with the learning rate set to 0.001. Fig. 3 showcases a graphical representation of the dynamic relationship between accuracy and the number of epochs for the Training and Validation phases. Fig. 4 depicts the progression of the Training and the Validation loss with the number of epochs. The version went through training for 25 epochs, a period regarded as required to accomplish optimum placement with the balanced dataset, therefore generating very positive results. Complying with the conclusion of 25 patterns the training procedure caused a very little training loss of 0.09 combined with a training accuracy of 97%. Precision together with recall metrics were gauged at 98.3% and also 97.64% specifically.

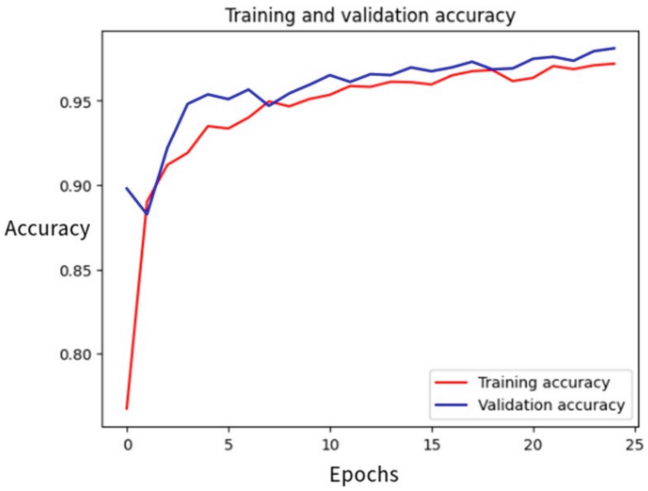


Fig. 3. Graphical representation showcasing the dynamic relationship between accuracy and the number of epochs across both the Training and Validation phases.

The validation loss was 0.067 with a validation accuracy of 97.2 percent and precision and recall were 98.7% and 96.5% respectively. The accuracy for the testing data of 390 images of each class was 96.5%, with a precision of 94% and a recall of 98%, respectively.

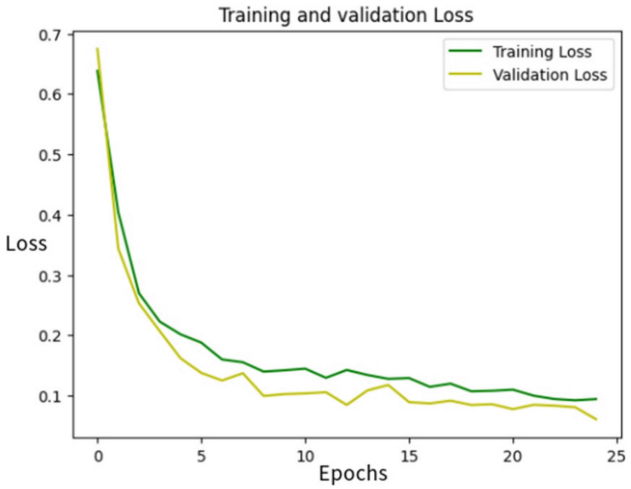


Fig. 4. The progression of the Training and the Validation loss with the number of epochs.

After the threshold optimization of the classifier i.e. the sigmoid function., with a threshold value of 0.68, the most optimized version of the model was achieved. With an accuracy of 96.5%, the precision and recall of 95% and 98.2% were achieved. The ROC curve, exhibiting an Area Under the Curve(AUC) of 0.98, portrays the classifier's remarkable discriminatory efficacy as evident from Fig. 5

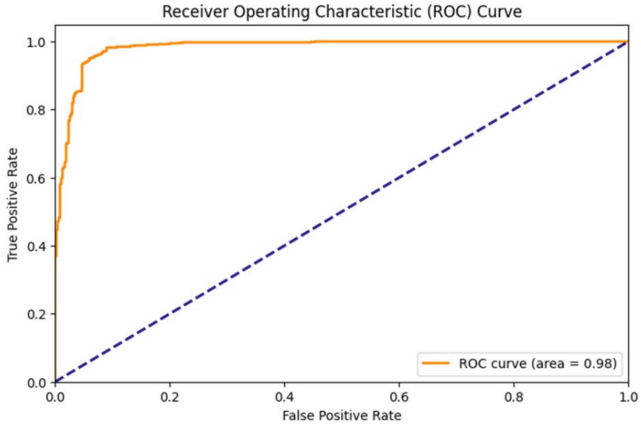


Fig. 5. Visual depiction of the True Positive Rate (TPR) versus False Positive Rate(FPR) curve.

5 Discussion

This analysis attempts to pit and rank two deep learning models, namely, VGG19, and ResNet50 against the custom CNN on different measures such as total parameters, training and validation losses, accuracy, precision, recall, and the F1- score of the training, validation, test data to give comparative results between the proposed model and other pre-trained models. The last part of this paper analyzes the performance of each of the models and the implications of each of the models. Fig. 6 shows a graphic representation of accuracy spanned across Training, Validation, and Testing phases for different Models. Fig. 7 is the visual depiction of Losses spanned across the Training, Validation, and Testing phases for different Models. The summarized details of different models trained using Transfer Learning is shown in Table 1.

Table 1. The summarized details of different models trained using Transfer Learning

Model	Total params	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy	Testing Loss	Testing Accuracy	Precision	Recall	F1 Score
Custom CNN	11M	0.09	0.97	0.067	0.972	0.2	0.965	0.950	0.982	0.966
VGG19	22.12M	0.091	0.966	0.101	0.960	0.270	0.910	0.753	0.628	0.685
ResNet50	31.97M	0.281	0.897	0.222	0.918	0.339	0.867	0.250	0.500	0.333

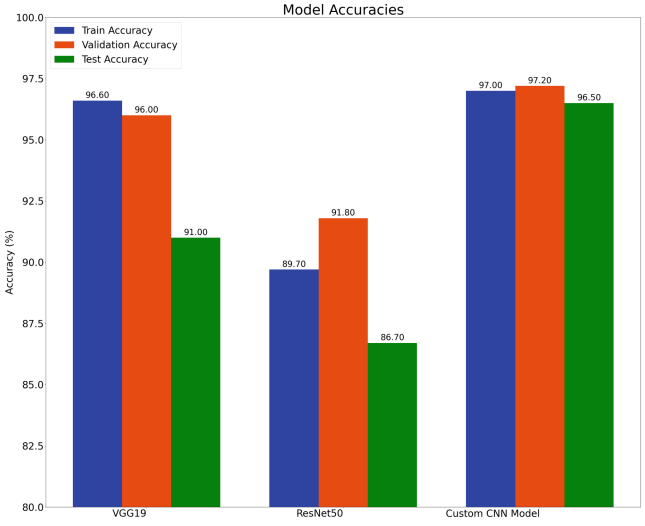


Fig. 6. Graphic representation of Accuracies spanned across Training, Validation, and Testing phases for different Models.

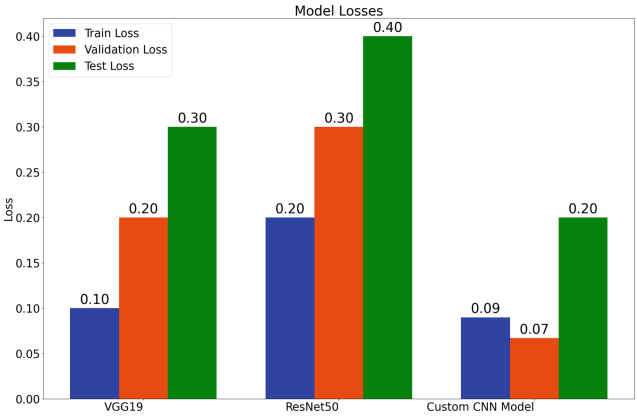


Fig. 7. Visual depiction of Losses spanned across Training, Validation, and Testing phases for different Models.

5.1 Vgg19

VGG19 comprises 22.12 million parameters. In the training and validation phase, the model performs well with a training accuracy of 96.6% as training accuracy and 96% as validation accuracy. That is, the testing performance could be more adequate, and the test accuracy achieved is only 91%. Nevertheless, regarding the accurate measure, VGG19 pulls out a high figure of 75.3% precision and an F1-score of 68.5%, and recall rates were scored at 62.8% causing it to miss some positive instances slightly. That is, the F1 scores for the training/validation set and the testing set are significantly different, which might imply the problem of overfitting.

5.2 ResNet50

The ResNet-50 model reached a training accuracy of 89.7% and had a loss of 0.281. Its validation accuracy was a bit higher at 91.8%, but we saw signs of overfitting because the training accuracy topped the validation accuracy. When it was on new test data, it scored an accuracy of 86.7% and a loss of 0.339. The model's precision came in at 0.250 On the flip side, its recall was 0.500 showing it caught half of the real positive cases. To give a balanced view of these numbers, we looked at the F1 score, which combines precision and recall. This score was 0.333.

5.3 General Observations

Overtraining is evident in all the models where there is a drop in the performance from the training/validation phase to the testing phase. Between the models, VGG19 is relatively faster with few parameters; hence, there is more importance on model architecture rather than just the parameters. Accordingly, it can be stated that none of the chosen metrics gives one model the upper hand over the other, which means that decisions have to be made on which aspects are more important when selecting the models.

6 Conclusion

In conclusion, our examination of pneumonia discovery uses deep understanding along with Convolutional Neural Networks and has noted a substantial enter clinical diagnosis. Leveraging a scrupulously curated data source with directory sites for normal and pneumonia situations, each additional fractional right into train examination together with recognition collections, including an astonishing 11 million parameters, has shown exceptional performance in the category.

The unique screening accuracy of 96.5 percent develops our version as a paragon in pneumonia discovery going beyond existing custom model versions. This accomplishment not only underscores deep discovery in clinical picture evaluation but additionally holds extensive effects for very early along with the precise medical diagnosis.

Our method not only showcases the ability of CNNs to recognize refined patterns within radiographic images but also vouches for the durability of our method through varied datasets. As we commemorate this success it is critical to recognize the continual advancement of modern clinical technology. Continuous initiatives ought to concentrate on improving our design, discovering added datasets, and addressing any type of intrinsic restrictions.

In the world of expert systems in healthcare, our study stands as a sign of development showcasing the exceptional abilities of deep understanding in pneumonia discovery. As we expect, this job not only adds to the academic discourse but holds assurance for real-world applications relocating us closer to a future where sophisticated modern technologies redefine the landscape of clinical diagnostics. One Step Forward, A Mile to Go. . .

Acknowledgments. We wish to acknowledge the support received from the Department of Computer Engineering, Dharmsinh Desai University for encouragement and for allowing the use of tools and resources necessary for the current work.

Disclosure of Interests.. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. McAllister, D.A., et al.: Global, regional, and national estimates of pneumonia morbidity and mortality in children younger than 5 years between 2000 and 2015: a systematic analysis. The Lancet Global Health **7**(1), e47–e57 (2019). https://doi.org/10.1016/S2214-109X(18)30408-X
2. Pandey, A., Galvani, A.P.: The burden of childhood pneumonia in India and prospects for control. The Lancet Child & Adolescent Health **4**(9), 643–645 (2020)
3. Waterer, G.: What is pneumonia?. Breathe **17**(3) (2021)
4. Singh, S., et al.: Pneumonia detection with QCSA network on chest X-ray. Scientific Reports **13**(1): 9025 (2023)
5. Al Mamlook, R., et al.: Investigation of the performance of machine learning classifiers for pneumonia detection in chest X-ray images. In: 2020 IEEE International Conference on Electro Information Technology (EIT). IEEE, (2020)

6. Bejan, C.A., Xia, F., Vanderwende, L., et al.: Pneumonia identification using statistical feature selection. Oxford Academic (2012)
7. Brajesh, K.: Convolutional Neural Networks: A Brief History of their Evolution. Medium (2021)
8. Saul, C.J., Urey, D.Y., Taktakoglu, C.D.: Early diagnosis of pneumonia with deep learning. arXiv preprint arXiv:1904.00937 (2019)
9. Hashmi, M.F., et al.: Efficient pneumonia detection in chest x-ray images using deep transfer learning. Diagnostics **10**(6), 417 (2020)
10. Pant, A., et al.: Pneumonia detection: An efficient approach using deep learning. In: 2020 11th International Conference on Computing, Communication and Net- working Technologies (ICCCNT). IEEE, (2020)
11. Rahman, T., et al.: Transfer learning with deep convolutional neural network (CNN) for pneumonia detection using chest X-ray. Applied Sciences **10**(9), 3233 (2020)
12. Lamia, A., Fawaz, A.: Detection of Pneumonia Infection by Using Deep Learning on a Mobile Platform. Computational Intelligence and Neuroscience 2022 (2022)
13. Akbulut, Yaman. "Automated Pneumonia Based Lung Diseases Classification with Robust Technique Based on a Customized Deep Learning Approach." Diagnostics 2 (2023): 260
14. Chiwariro, R., Wosowei, J.B.: Comparative analysis of deep learning convolutional neural networks based on transfer learning for pneumonia detection. Int. J. Res. Appl. Sci. Eng. Technol **11**(1), 1161–1170 (2023)
15. Xue, X., et al.: Design and Analysis of a Deep Learning Ensemble Framework Model for the Detection of COVID-19 and Pneumonia Using Large-Scale CT Scan and X-ray Image Datasets. Bioengineering **10**(3), 363 (2023)
16. Kermany, D., Zhang, K., Goldbaum, M.: Labeled optical coherence tomography (OCT) and chest X-ray images for classification (2018). Mendeley Data, v2 https://doi.org/10.17632/rscbjbr9sj https://nihcc.app.box.com/v/ChestXray-NIHCC (2018)
17. Temraz, M., Keane, M.T.: Solving the class imbalance problem using a counterfactual method for data augmentation. Machine Learning with Ap- plications **9**, 100375 (2022)
18. Xu, M., et al.: A comprehensive survey of image augmentation techniques for deep learning. Pattern Recognition 109347 (2023)
19. Vogelsang, D.C., Erickson, B.J.: Magician's Corner: 6. TensorFlow and TensorBoard. Radiology: Artificial Intelligence **2**(3), e200012 (2020). https://doi.org/10.1148/ryai.2020200012
20. Krichen, M.: Convolutional neural networks: A survey. Computers **12**(8), 151 (2023)



Leveraging Cloud Resources for Machine Learning-Based Spam Detection API

Ashish Revar[1(✉)], Bhavin Bhesaniya[2], and Kalpesh Wandra[3]

[1] SITAICS, Rashtriya Raksha University, Gujarat, India
ashish.revar@rru.ac.in
[2] Research Scholar, SITAICS, Rashtriya Raksha University, Gujarat, India
22mcsdf006@student.rru.ac.in
[3] Pro Vice Chancellor, Rashtriya Raksha University, Gujarat, India
kalpesh.wandra@rru.ac.in

Abstract. The proliferation of spam emails continues unabated and challenges the security of digital communication systems, thus necessitating robust detection mechanisms. This paper develops an exhaustive spam detection system based on cloud computing resources as well as advanced ML models. The ability of AWS to scale up to take on a variety of data makes it perform real-time spam detection. The central contribution is a hybrid approach of ML based on logistic regression, SVM, random forests, and LSTM. This can address challenges including but not limited to scalability, adaptability to ever-evolving spam patterns, as well as real-time processing. Detailed evaluation and analysis show vast improvement in terms of detection accuracy and scalability at higher levels, which makes it a very effective solution for modern detection challenges against spam. Integration with cutting-edge security tools will help the system combat all possible threats.

Keywords: Cloud Computing · Machine Learning · Cybersecurity · Spam Detection · Scalability · Real-time Processing · Deep Learning

1 Introduction

Spam emails remain one of the most alarming and dynamic threats to digital communication systems. Apart from disrupting private and professional communications, spam messages are also carriers for malware as well as phishing and other cyber-attacks. The dynamic nature of spam requires the continuous development of robust detection mechanisms beyond traditional rule-based systems based on static heuristics. Due to their inability to encompass the complexities and changing strategies used by evolved modern spamming attacks, these legacy systems often raise high false positives and false negatives [10, 11]. The evolving nature of spam tactics places high constraints on the adaptability of traditional rule-based systems, thus demanding better and scalable solutions.

Previous studies have shown the use of various ML algorithms to be implemented to enhance the process of spam detection. The algorithms are thus able to improve the

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 305–325, 2025.
https://doi.org/10.1007/978-3-031-86296-0_22

detection accuracy while facing important scalability and real-time processing challenges. According to Sultana (2020) [10], this is because the classic ML techniques have problems with the large scale and their dynamic nature in collecting emails, which continuously evolve with new spam patterns. Besides, Kumar et al. (2020) [11] report that the current ML models tend to struggle for high precision and recall, mainly because the preprocessing and feature extraction methods applied in such models are incapable of expanding their functionalities in real-world applications. Consequently, a scalable spam detection system with real-time effectiveness is one of the essential requirements in state-of-the-art cybersecurity practice.

To circumvent these limitations, this paper discusses a more advanced spam detection system, capitalizing on the computational powers of scalability brought about by cloud computing. The authors make use of the structural backbone of the infrastructure, Amazon Web Services (AWS), to develop a system capable of processing vast amounts of email data in real-time. The system itself dynamically adapts according to changing loads, thereby maintaining scalability as fluctuations in the volume of email traffic. Our approach incorporates a variety of state-of-the-art ML algorithms, such as logistic regression, support vector machines (SVM), random forests, and deep learning models like Long Short-Term Memory (LSTM) networks. These algorithms have been chosen and optimized carefully, by taking into consideration the challenges of scalability and detection accuracy presented in previous research work [2–4].

This apart, advanced data preprocessing techniques were integrated to enhance the precision of the system under consideration. Here, we have made use of tokenization, stop-word elimination, and Term Frequency-Inverse Document Frequency (TF-IDF) coupled with word embedding techniques to enhance the feature extraction, which plays a vital role in the performance of any machine learning algorithm used in spam detection. By cleaning the input data, models were further trained on differentiating spam from legitimate emails more effectively thus increasing the rate of detection and lowering false positive percentages.

In conclusion, this paper introduces the very first integration of cloud-based infrastructure and modern ML algorithms for an existing gap in spam detection systems.

Our approach not only provides scalability and real-time solutions but also shows enormous improvements in terms of accuracy and adaptability towards new and evolving spamming techniques. This research works as a contribution to the wider domain of cybersecurity through an effective, robust, and efficient spam detection system that can work on dynamic large environments.

2 Related Work

2.1 Machine Learning in Email Spam Detection

Machine learning has dramatically influenced the development of techniques to detect spam in emails. In the 2004 introductory work by Goodman et al. was Bayesian filter, formed as the basis for further development in the field [1]. Almeida et al. showed that SVM and neural networks improved the accuracy more than a traditional system of rule-based ones through better detection of more sophisticated spam patterns [2]. Zhang et al. extended this work by using deep learning models, particularly CNNs, which

effectively solved the problem of image-based spam detection, pushing the frontiers of spam classification even further [3]. Malhotra et al. in their attempt to use LSTM to capture the temporal behavior in the sequence of emails, thereby enhancing the rate of detection by taking into account the flow of information in messages [4]. While these machine learning approaches improved the detection accuracy, implementing them on large-scale datasets, especially in real-time, remained a challenge.

2.2 Cloud Computing for Scalability

Cloud computing platforms such as Amazon Web Services, Google Cloud, and Microsoft Azure are strong in scalability and widely used in spam detection systems. Armbrust et al. (2010) described the benefits of cloud scalability related to dataintensive operations, which is especially relevant for large-scale spam filtering systems, particularly for decreasing latency and computational expense [5]. Wu et al. (2014) demonstrated further that cloud-based solutions can process massive datasets in realtime and are therefore well-suited for applications requiring computational power within challenging times like spam detection [6]. Chang (2020)[16] gave a comprehensive framework for cloud adoption in cyber security applications and pointed out how scalable, flexible, and low-cost solutions with great potential for managing the increasing complexity of spam detection were offered with the services of the cloud. Despite the clear advantages, cloud-based implementations of spam detection systems still face integration challenges when combined with machine learning models.

2.3 Hybrid Approaches

The merger of the cloud computing and machine learning concept fills up gaps left by autonomous systems. In this line, Tsai et al. (2019) proposed hybrid models to introduce the cloud with continuous updates on machine learning to keep in tandem with the evolving spammer techniques [8]. Li et al. (2020) proposed an approach that integrates cloud infrastructure with complex ML models to facilitate more effective and scalable spam filtering for massive datasets [9]. These hybrid approaches tap into the real-time computational processing of the cloud and the intricate pattern recognition capabilities of the ML model to provide powerful yet adaptive systems for spam filtering.

2.4 Recent Studies and Advances

The integration of cloud computing and machine learning further develops recent research on the aspect of spam detection. According to Sultana (2020) [10], the overhead of scalability and efficiency of the spam detection system can be heightened by the cloud infrastructure. Kumar et al. (2020) [11] pointed out the accountability of cloud resources in upgrading performance and scaling features of machine learning algorithms, mainly in high-throughput environments of spam detection. Wolf et al. (2020) also introduced the transformer architecture, which can very well do NLP tasks and shows its use in the classification of spam e-mails using a language model approach [12]. Jain and Sharma (2022) made a comparison of different machine-learning algorithms that can be used for

spam detection. That paper emphasized ensemble methods used for spam detection and described that preprocessing the data leads to great effects in improving the model [13]. Ali and Lokhande further advance the productivity aspect of feature engineering and preprocessing, underlining the significance of such procedures for the efficient formulation of machine learning models in spam detection [14].

2.5 Comparative Analysis and Research Gaps

Several gaps in research in the area and important insights are revealed in a comparative study. The first machine learning techniques that tried to lay the foundation for spam detection were Bayesian filters and SVMs which failed mostly on scalability and real-time processing abilities. Deep learning models, such as CNNs and LSTMs, tried working over some of those limitations by improving accuracy as well as handling more complex patterns of spam. However, these models are very computationally power-intensive and challenging to use on a large scale in real applications.

Cloud computing platforms can provide the infrastructures needed for large-scale spam detection by the deployment of machine learning models. However, one of the challenges arising out of such advanced cloud computing that integrates the model of machine learning is the question of how such systems could keep abreast with fast-changing approaches in spam techniques and yet ensure efficiency and accuracy.

Based on such above analysis, our research addresses such gaps by developing a cloud-based spam detection system relying on advanced machine learning algorithms as well as robust data preprocessing techniques. The cloud-based system dynamically scales up or downloads and provides good real-time spam detection accuracy through the application of cloud resources. It is integrated to enhance the performance of the detector while helping to circumvent the scalability and adaptability problems as identified in previous studies.

3 System Architecture

The proposed system architecture consists of cloud infrastructure and advanced machine-learning models that enable real-time spam detection. The system architecture has been designed considering its limitations, specifically its scaling difficulties, huge volumes, and sensitivity towards time, all of which were issues. Some examples include large email datasets. The proposed system would be based on a cloud-based architecture; the system would scale dynamically, and emails would be processed in large quantities to guarantee high-accuracy detection of spam.

- **User Interface:** The system features a Django web interface. This enables the users to interact with the spam detection system. Users can submit emails for spam detection and get results in real time. This user interface is responsive and ensures that users receive direct feedback, which is important for real-time spam detection in large-scale systems.

- **API Layer:** A whole layer of RESTful API was built with "Django-restframework," thus taking care of all communication that needs to occur between the user interface and machine learning models. This API processes incoming email requests, applies if needed preprocessing, and returns the spam classification results to the user. This API is designed with high availability and reliability in mind, leveraging AWS cloud infrastructure to ensure seamless scaling during high-traffic periods.
- **Machine Learning Models:** The system uses several machine learning models, mainly Logistic Regression, Support Vector Machine (SVM), Random Forest, and Long Short-Term Memory networks, trained on large, labeled email datasets such as spam and ham. The breadth of these models ensures a balanced spam detection system since each model carries out the overall accuracy and adaptability of the system based on the changing nature of the spam pattern. The models are periodically retrained using new datasets to maintain high accuracy and adapt to the ever-changing nature of spam emails.
- **Cloud Infrastructure:** Its hosting will utilize AWS EC2 instances, which provide the necessary computation to the system to be able to carry out real-time spamming detection. Docker containerization ensures consistent deployment between environments. It allows for dynamic scalability as email traffic picks up with the orchestration of its deployment, scaling, and managing of ML models. AWS S3 is utilized for scalable data storage, while AWS RDS ensures the handling of relational database management. This makes it possible for the system to process big data efficiently while handling any high-volume emails in real-time.
- **Data Preprocessing:** It is then followed by the preprocessing stage, whereby the emails feed ML models. This includes tokenization, HTML tag removal, stop-word removal, and vectorization. Techniques such as Term Frequency-Inverse Document Frequency and word embedding are applied to convert raw content for the emails into structured data that can be interpreted by machine learning models. Quality of preprocessing is necessary for improving model performance because it absorbs noise and reduces the same along with extracting only relevant features for the classification.
- **Data Pipeline and Continuous Integration:** It is implemented as a CI/CD pipeline using GitHub Actions, automatically introducing new data and updating ML models. Such a pipeline includes automated testing and monitoring to ensure that changes introduced into the models or data do not compromise the reliability or accuracy of the system in any form. This CI/CD pipeline is crucial since new spam patterns arising from the evolving nature of spam need to be incorporated into the training datasets to maintain the effectiveness of the system.
- **Evaluation and Monitoring:** The system continuously undergoes monitoring on predefined performance metrics, including accuracy, precision, recall, and F1-score. The system is evaluated based on benchmark datasets, including the SpamAssassin Email Classification Dataset, and also custom datasets collected from user feedback. The scalability of the proposed system under real-time processing will be tested in different scenarios with variable loads while giving attention not to reducing the email classification accuracy significantly.

3.1 System Workflow

To handle large email datasets, the system follows a modular workflow combined with real-time spam detection and user feedback. The main steps of this workflow are as follows:

- **Data Collection:** Emails are accumulated continuously and stored in AWS S3 buckets. The S3 service provides consistency and scalability in performing storage with the bigger volume of emails the system is processing, making it easier to retrieve and easily integrate into the ML models so that data processing happens more effectively.
- **Preprocessing:** All incoming e-mails are preprocessed using custom Python scripts that clean and normalize the data. The preprocessing steps involve tokenization, removal of stop-words, and vectorization using TF-IDF or word embeddings. These transforms are necessary to preprocess raw email text in a structured format which the machine learning model can process efficiently.
- **Model Training and Retraining:** The preprocessed email data is then utilized for training several machine learning models. The cross-validation technique as well

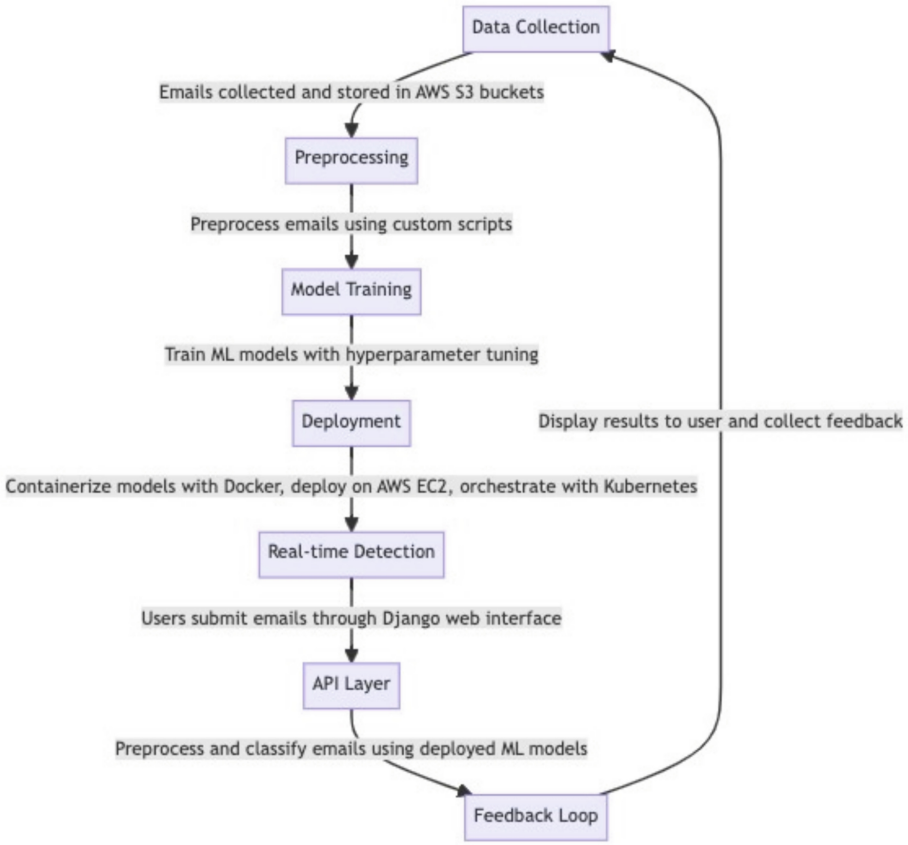


Fig. 1. System Architecture for Real-time Spam Detection

as hyperparameter tuning are applied to optimize the performance of the model. A retraining mechanism also exists in the system, whereby models are updated with fresher data based on a specified schedule so that it is fit enough to detect the evolving spam techniques with good accuracy. In this way, the system continues to be effective with time (Fig. 1).

- **Deployment:** Trained models are containerized by Docker and deployed on AWS EC2 instances. The deployment process is handled by Kubernetes so that the whole system scales up and down automatically to deal with all incoming emails, meaning that the system's performance is at high levels even during very heavy traffic in the times of email.
- **Real-time Detection and User Feedback:** Users interact with the system through a Django web interface that accepts emails to be classified into spam and ham. The API layer handles these requests. Using deployed models of machine learning, it classifies the emails submitted into spam and ham. The system returns feedback immediately to users: within a second of them submitting their email, the system will post the result of classification. A feedback loop has been established through which the system can get feedback on its accuracy in classifications, thus allowing further refinement of its performance.
- **Feedback Loop and Continuous Improvement:** The system involves a feedback loop where the users may raise false positives or false negatives about the classification results. Concerning the given feedback, the models used are retrained to improve their precision and adaptability concerning new spam techniques. This is essential in keeping the system updated, on the changes in the pattern of spamming over time.

4 Methodology and Implementation

This section discusses the collection of data, preprocessing the data, developing a model, and finally implementing the system. In an attempt to see into the practical applications for machine learning models on spam detection, along with deploying them on cloud infrastructure and developing a user-friendly interface that will perform real-time spam detection, such work is studied.

4.1 Data Collection and Preprocessing

For training and testing the spam detection models, publicly available datasets-Enron Email Dataset, SpamAssassin Public Corpus-were used. These datasets provided labeled examples of spam emails, along with corresponding ham emails. Preprocessing is quite crucial. Raw email data needs cleansing from noise since subject content in emails generally carries lots of noise. The important preprocessing techniques among them are:

- **Text Normalization:** All email text is converted in lower case for standardization.
- **Tokenization:** The text of the email is tokenized as individual words or tokens.
- **Stop-word Removal:** All common words that carry no spam detection value like"and,""the," etc., have to be filtered off.

- **Feature Extraction:** The above preprocessed text is then converted into numerical features using Term Frequency-Inverse Document Frequency (TF-IDF) and word embeddings, so that it can be fed into the model.

According to AbdulNabi and Yaseen (2021),"appropriate preprocessing is required to improve accuracy in spam detection models as it allows machine learning algorithms to focus on meaningful patterns in the data rather than on the noise".

4.2 Model Training and Evaluation

Training many different variants of machine learning models over the processed data: For each one of these, hyper-parameters were tuned through experimentation. The models are scored with a set of evaluation metrics that mainly comprises accuracy, precision, recall, and F1-score. For machine learning algorithms trained for spam detection:

- Logistic Regression
- Support Vector Machine (SVM)
- Random Forest
- Naive Bayes
- Long Short-Term Memory (LSTM) Networks
- Hugging Face Transformer Models

The distinctive strengths and weaknesses of each of the algorithms considered were explored in light of specific requirements for spam detection, including real-time classification, scalability, and robustness to new spam patterns.

4.3 API Development

A Django-based REST API was developed to serve the inbound requests for spam detection. The API endpoints enable communication from the user interface to the backend machine learning models. The API can perform actions such as email preprocessing, applying machine learning models, and delivering the prediction. Different security characteristics, including authentication using JSON Web Token (JWT) and rate limiting, have been implemented to limit access to only the appropriate users of the API as a means of prevention against denial-of-service attacks.

4.4 Frontend Development

The user interface was developed as an extension of the code for HTML and CSS with JavaScript to ensure this email submitting process is simple with actual real-time results on spam detection. Front-end communication with the backend API is applied to bring about integration without interference into the system's speeds and efficiency of response to users and results access.

4.5 Machine Learning Models and Mathematical Formulations

Our spam detection tool uses a combination of several machine learning algorithms, each because of its ability to address the different complexities presented by the email dataset. For the role they play in this spam detection, advantages, and potential shortcomings in spam detection, we provide a more specific discussion about each algorithm.

4.5.1 Logistic Regression

Logistic Regression is an algorithm that describes the probability of an email being classified as spam or not based on the characteristics of that e-mail. The logistic function, defined as:

$$P(y = 1|X) = \frac{1}{1+e^{-(\beta_0+\sum_{i=1}^{n}\beta_0 X_i)}}$$

Where:

- y is the binary outcome (1 for spam, 0 for not spam),
- X is the feature vector extracted from the email,
- β_0 is the intercept, and
- β_i are the coefficients for each feature X_i.

Benefits:

- **Fast and Simple:** Logistic regression is computationally efficient making it a good candidate for real-time spam detection.
- **Interpretable:** Coefficients of the model provide information on the features best describing spam.

Drawbacks:

- **Assumes Linearity:** Assumes that the relationship between features and spam probability is linear, probably not so with spam patterns.
- **Not Ideal for Complex Spam:** The model would miss to identify sophisticated spam patterns that are based on non-linear relationships.

4.5.2 Support Vector Machine (SVM)

SVM classifies the email as spam or not spam by constructing a hyperplane in that high-dimensional feature space. The feature space is achieved from the extracted characteristics such as word frequencies and email metadata and any other distinguishing features indicative of spam. Therefore, the SVM separates spam from non-spam messages by

maximizing the margin between the two classes. The decision function is represented as:

$$f(X) = \sin n(w \cdot X + b)$$

where:

- w is the weight vector, determines the direction of a separating hyperplane.
- b is the bias term. This transforms the hyperplane into maximizing separation between spam and non-spam e-mails.

The optimization problem for maximizing the margin is defined as:

$$min \frac{1}{2} \|\omega\|^2 \text{ subject to } y_i \, (w \cdot \, X_i + b) \geq 1 \forall i$$

where y_i is the class label (spam or non-spam) and X_i is the feature vector for the i-th email.

Benefits in Spam Detection:

- **Effective for High-dimensional Data:** SVM works very efficiently with large feature sets of data, which are typical in email collections. Metadata and word occurrences enable an SVM to work effectively in a large high-dimensional space.
- **Robust Separation:** SVM really does a great job finding spam patterns because it properly creates separation between the spam or not emails using some complex relationships between attributes, creating a separation boundary sometimes called a hyperplane between spams and nonspams.

Drawbacks in Spam Detection:

- **Computational Intensity:** As the email data is truly humongous, SVM will be computationally intensive. Further, extracting features involves a lot of complexity, which makes it pretty burdensome in nature. Therefore, in real-time spam detection systems where efficiency is an issue, it presents a problem.
- **Struggles with Overlapping Data:** SVM is prone to fail when the features of spam emails and non-spam emails show considerable overlap; SVM will likely misclassify. The efficiency of SVM will be inhibited when spam indicators are not adequately separable from noise datasets.

Role in Spam Detection: Using spam detection, SVM makes an excellent classifier for email classification if features clearly distinguishing spammers from nonspammers are appropriately defined and of a high dimensionality. SVM is well suited for the complicated datasets in email collections due to its robust capability in dealing with large and complex feature sets. However, there is a need to manage computational constraints and to reduce noise in data to achieve optimal performance. SVM is used as part of our spam filtering system and will provide the basis for an accurate classification, although a number of filtering techniques might need to be applied in order to clean noisy data and overlapping or sparsely distributed events.

4.5.3 Random Forest

Random forest is an ensemble learning method that creates multiple decision trees and trains each on a sample of the email data to improve accuracy in classification. It is apt to the complexity of patterns with multiple trees used to make a decision. The prediction function is:

$$f(X) = \frac{1}{T} \sum_{t=1}^{T} h_t(X)$$

Where:

- T is the number of trees, and
- $h_t(X)$ is the prediction of the t-th tree.

Benefits in Spam Detection:

- **Reduces Overfitting:** This is because Random Forest reduces overfitting by averaging prediction from different decision trees so it will make a very good real-time spam detection, even if the datasets in question are very different.
- **Feature Importance:** Random Forest can produce a feature importance ranking, so you can then see which features were most informative for spam detection (e.g. words in the text or metadata relating to the e-mail).
- **High Accuracy:** The proposed approach achieves high accuracy in the spam detection setup by making use of Random Forest, that uses the ensemble learning nature, especially by handling diverse spam patterns.

Drawbacks in Spam Detection:

- **Computational Overhead:** The process of training and maintaining an ensemble of decision trees is computationally expensive, even in cloud-based spam filtering systems.
- **Reduced Interpretability:** There is better accuracy of Random Forest but somewhat at the expense of interpretability compared to a simpler model like decision trees that makes this harder to understand individual predictions.

Role in Spam Detection: Random Forest is also effective for detecting complex, varied spam patterns in large datasets, increasing the robustness and accuracy of the spam classification by its integration of multiple decision trees.

4.5.4 Naive Bayes

Naive Bayes classifiers apply Bayes' theorem with the assumption of conditional independence of all features. In spam detection, it calculates the probability that a given

e-mail is spam based on features such as word frequency, information in headers, and structure of the e-mail. The formula for Naive Bayes is:

$$P(c|X) = \frac{P(X|c) \cdot P(c)}{P(X)}$$

where c is the class (spam or non-spam) and X is the feature vector for the email.

Benefits in Spam Detection:

- **Fast and Efficient:** Naive Bayes is highly computationally light weight, which makes it suitable for spam detection systems that handle very large volumes of e mails in real time.
- **Good for Simple Patterns:** It performs well in scenarios where the spam patterns are relatively simple and are identifiable by word frequency or header information.

Drawbacks in Spam Detection:

- **Feature Independence Assumption:** Naive Bayes assumes that the features, for instance, the words in an email message, are independent of each other. In practical situations, this independence could rarely be attained with email data sets and that may reduce its effectiveness in more subtle spam.
- **Limited for Complex Spam:** The model has other drawbacks since it cannot capture dependencies between features of complex spam techniques hence its ability to detect modern spam patterns is easily diminished.

Role in Spam Detection: Naive Bayes is very efficient as a baseline model of spam detection, particularly useful for the purpose of first-round filtering of simple spam patterns; though often used in conjunction with additional methods to cope with more complex types of e-mails.

4.5.5 Long Short-Term Memory (LSTM)

These, of course, are a variant of Recurrent Neural Networks (RNNs), so perfectly suited to long sequences: They do capture long term dependencies so well. Specifically, LSTM networks are useful for recognizing patterns that come over time in the word sequence or phrase sequence. The fundamental update functions are:

$$\begin{aligned} i_t &= \sigma\left(W_i \cdot \left[h_{t-1}, X_t + b_i\right]\right) \\ f_t &= \sigma\left(W_f \cdot \left[h_{t-1}, X_t + b_f\right]\right) \\ O_t &= \sigma\left(W_o \cdot \left[h_{t-1}, X_t + b_o\right]\right) \\ C_t &= f_t \times C_{t-1} + i_t \times C_t \end{aligned}$$

Benefits in Spam Detection:

- **Captures Temporal Dependencies:** LSTM appears to have a very strong capability for capturing long-range dependencies in sequences of emails, such as repetitive spam phrases or temporal patterns in the behavior of emails.

- **Contextual Understanding:** LSTM can understand relationships between words or phrases over an entire email and hence tends to be effective in detecting sophisticated patterns of spam that are based on nuanced contextual information. **Drawbacks in Spam Detection:**
- **Computationally Intensive:** LSTM would require massive amounts of computations for the large dimensions, such as huge datasets from emails in training-the model, and hence may not be an ideal candidate for real-time spam detection unless deployed on scalable cloud infrastructure.
- **Complex Architecture:** Training an LSTM network is known to introduce longer times because it has a complexity within which to be trained; hence, prompt deployment in fast-changing spam detection environments becomes difficult.

Role in Spam Detection: LSTM is very good at picking spam that will exploit temporal or sequential patterns in email. In point of fact, it is especially useful when the spam varies with time or has cycle patterns that generally get missed by relatively simple models.

4.5.6 Hugging Face Transformer Models

Transformers can utilize self-attention mechanisms to automatically identify the relationships between words in an email, irrespective of how far apart they may be in the text. This strength makes transformers very successful in identifying complex spam patterns that were rather missed by the previous models. The scaled dot-product attention mechanism is:

$$AttentionQ|K|V = softmax^{\left(\frac{QK^T}{\sqrt{d_k}}\right)}$$

where:

- Q, K, and V are the query, key, and value matrices of the input.
- d_k is the dimension of the key vectors.

Benefits in Spam Detection:

- **Superior for Long-range Dependencies:** Transformers are quite effective at detecting subtle patterns of dependencies between words or phrases and can be very good at finding subtle spam patterns that spread over an entire email.
- **Nuanced Pattern Detection:** Transformers can find the subtle patterns spammers use, which include minor manipulations in the usage of words, metadata, or structure that earlier models could not detect.

Drawbacks in Spam Detection:

- **Computationally Demanding:** Transformers consume extensive computational powers and memory, especially when dealing with large datasets. Therefore, it is likely costly to deploy in actual spam detection systems that have to react in real time.
- **Longer Inference Times:** Complexity of transformers may result in longer processing times; hence, if not optimized, this becomes a limitation for deploying it in a real-time spam detecting system.

Role in Spam Detection: Other than NLP models provided in Hugging Face, the transformers they provide can detect complex patterns of spam with high performance based on a contextual meaning of the words. These capabilities make them effective tools in dealing with advanced spam that even evade the usual traditional methods of spam detection.

5 Performance Evaluation

This section evaluates the system in terms of accuracy, precision, scalability, security, and cost-effectiveness. These metrics are crucial to prove that the suggested spam detection system is feasible and can be applied in real time on cloud infrastructure.

5.1 Accuracy and Precision

It measures precision and accuracy through a test dataset. Comparison of the performance of implemented machine learning models is done using accuracy, precision, recall, and F1-Score. In all the models, the Random Forest model produced the highest accuracy as 97.55% and precision as 98.57%. Table 1 shows comparative performance analysis of the models in the following pages:

Table 1. Performance Evaluation of Machine Learning Models

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Logistic Regression	95.20	94.30	96.00	95.10
SVM	97.00	96.10	97.30	96.70
Random Forest	97.55	98.57	97.00	97.78
Naive Bayes	93.10	92.30	93.50	92.90
LSTM	99.00	98.80	99.10	98.95
Transformer	99.10	99.00	99.20	99.10

From the results obtained, while LSTM and Transformer models perform well, Random Forest is rather a balanced model in its terms of both accuracy and computation. High performance by the Random Forest is due to ensemble learning that reduces overfitting and improves generalization ability from data.

5.2 Scalability and Load Testing

To test the scalability of the system, concurrent users were simulated. AWS autoscaling features maintained the increasing workload to ensure constant performance under varying loads. Apache JMeter was used for simulating up to 1,000 concurrent users submitting email spam detection requests. Table 2 sums up the system's response times, throughput, and error rates:

Table 2. Load Testing Results

Concurrent Users	Average Response Time (ms)	Throughput (requests/sec)	Error Rate (%)
100	120	8.3	0.0
500	150	28.1	0.1
1000	200	45.2	0.3

It shows that such a system can support up to 1,000 concurrent users with an average response time of 200 ms with an error rate of 0.3%. The majority of the performance is donated by Docker for containerized functionality and Kubernetes for orchestration. Docker implies that the many parts of a system will function independently inside isolated environments, without resource conflicts and bottlenecking. This will dynamically scale an application on the basis of load to automatically ensure high availability as well as efficient management of resources.

5.3 Security Testing

Since vulnerabilities may be present, making the spam detection system well-secured is vital. The system was subjected to tests using various security testing tools to identify and mitigate security risks. For SAST, SonarCloud was applied for source code analysis to spot any vulnerabilities already existent in the application. For SCA, Snyk was applied for finding issues in third-party dependencies. Lastly, the Dynamic Application Security Testing (DAST) was performed using OWASP ZAP, simulating real-world attacks on the application under deployment.

Table 3 presents the results of security testing:

Table 3. Security Testing Results

Tool	Type	Issues Detected
SonarCloud	SAST	5 minor vulnerabilities
Snyk	SCA	2 critical, 3 high, 8 medium
OWASP ZAP	DAST	1 medium, 4 low

The vulnerabilities were addressed promptly, enhancing the system's overall security posture. JWT token-based authentication and rate limiting ensure safe access to the API, prevent possible unauthorized accesses and DoS attacks.

5.4 Cost Analysis

The cost of deployment and running on AWS was calculated, taking into account key factors like EC2 instances, S3 storage, RDS database, and data transfer. The monthly cost breakdown is illustrated in Table 4:

Table 4. Estimated Monthly Costs for AWS Resources

Resource	Monthly Cost (USD)
EC2 Instances	250
S3 Storage	50
RDS Database	100
Data Transfer	30
Total	**430**

Using Infrastructure-as-Code (IaC) tools such as Terraform, the resources required by the system were efficiently managed, and auto-scaling ensured that costs remained proportional to usage. Terraform automates the provisioning, scaling, and management of cloud resources, minimizing operational overhead while maintaining cost efficiency.

Infrastructure-as-Code (IaC) tools such as Terraform helped manage the system resources that were required and kept costs proportionate with usage due to autoscaling. The Terraform services in cloud computing take care of provisioning, scaling, and managing the resources, which thereby reduces the operational overhead while maintaining cost efficiency.

5.5 Summary of Performance Evaluation

Effectiveness, scalability, security, and cost efficiency are some of the key performance metric indicators. Under this vision, the Random Forest model promises to give the best trade-off between the performance and the cost of computation. Benefits include the fact that cloud infrastructure is used, whereby the designed system can easily take heavy loads without a drop in efficiency. The advanced security features increase the resiliency of the system to handle possible threats. Overall, the system boasts a very good achievement of the desired performance criteria for state-of-the-art spam detection systems.

6 Security Measures

This section sheds light on the robust security mechanisms in place in protecting user data and securing the API endpoints of the system. It aims to be assured of data confidentiality, integrity, and availability while combating vulnerabilities.

6.1 Data Security

The system ensures data security through the use of industry standard encryption protocols and secure techniques for data storage. To ensure protection for user credentials, the system uses the Bcrypt hashing algorithm in the storage of passwords. The password-guessing security with the combination of salts and a computational cost factor resists brute force attacks from successful attempts to guess passwords and may be able to guess passwords in practice if the server is not properly optimized. The system is not only password-protected but also uses Transport Layer Security for encrypted data in transit, meaning that there is no way to intercept or alter the data while in transit. To further protect sensitive information on the server, the server uses Advanced Encryption Standard 256 for encrypting data at rest. AES-256 is one of the most secure ways of encryption where the stored data cannot be accessed by anybody else except the owner even if there is a breach in security. All these methods together ensure that the data transferred from the system is safe as well as well-safe when stored on the system.

6.2 API Security

The API endpoints of the system require safe integrity and reliability in delivering the spam detection service; consequently, JWT is applied for authentication purposes to issue a JWT for use on subsequent requests to access the API each time a user successfully logs into their account. This will use a secure secret key to generate a signed JWT, and thus only the authorized users will be able to communicate with the system. This method enables support for lightweight authentication in a stateless way, thus leading to a highly scalable application for cloud-based applications.

The API rate limiting is used to prevent the abuse of the API as well as Denialof-Service attacks. With rate limiting, the number of requests that can be made by a user within a given time period is curtailed; therefore, it minimizes conditions that could lead to an overload in the system due to the excessive requests. The security measure ensures that even at times of maximum demand, the system remains responsive and available.

In addition to these preventive steps, the system tests its security periodically to discover and eliminate any hidden vulnerabilities. Three major security testing approaches are used:

- **Static Application Security Testing (SAST):** SonarCloud is scanning the source code for continuous detection of such security vulnerabilities as SQL injections, cross-site scripting (XSS), and dangerous code practices.
- **Software Composition Analysis (SCA):** This application determines known vulnerabilities existing in third-party dependencies by using Snyk, such that the libraries and frameworks involved within the system are secure and updated.
- **Dynamic Application Security Testing (DAST):** It will then simulate real attack conditions through scanning the live API for vulnerabilities such as cross-site request forgery, insecure API endpoints, and input validation flaws.

With the combination of SAST, SCA, and DAST, keeps monitoring the system, thereby hardening against new and emerging threats. It uses the outcomes of these tests with some security patches and API updates, making it secure for a long time.

The multifaceted security allows protection from as many attacks as possible through multiple vectors, finally culminating in self-protection of the system.

7 Future Work

A few major areas for which enhancement is recommended have been identified to further improve the effectiveness and adaptability of the spam detection system. These enhancements will include bringing more evolved technologies into the system, providing more flexibility for the users, and ensuring continuous improvement of the system with the aid of real-world feedback. The key issues identified are as follows:

- **Integration of Advanced Threat Intelligence:** Future versions of the system will incorporate real-time feeds of threat intelligence feeds obtained from cybersecurity organizations and public repositories, which will contain recent releases regarding emerging threats and spam techniques. Since this feed is incorporated automatically into the spam detection models, the system will automatically respond to new attack patterns dynamically, improving its detection rate and not relying on manual intervention. The machine learning algorithms applied will be able to continue learning from these intelligence sources; hence, the system will remain ahead of rapidly evolving spam tactics.
- **Customizable Filtering Rules:** The current systems of spam detection are also shown to be static and non-flexible when it comes to dealing with the end users. Future changes in the system will give the user the choice to implement and update filtering rules based on individual demands. The user can make specific filters by blocking keywords, email addresses, or particular domains. This would be highly helpful for the organizations whose email traffic has a unique pattern or requires more granular control over what constitutes spam. And permitting flexibility to customize filtering rules will definitely improve user satisfaction by controlling the filtering process more intensely.
- **Enhanced Reporting and Analytics:** The availability of more detailed reporting and analytics tools will enable the users to have a clear view of the systems' behavior in detail. Some things that this can entail include tracking time trends for spam detection, identifying particular types of spam that most often bypass filters, and monitoring the overall accuracy of the system when operated in different scenarios. Advanced data visualization tools will be able to view real-time and historical views, which allows for better proactive management of their email security. Analytics will enable better identification of weak points in the email infrastructure and consequent policies amendments of organizations.
- **User Feedback Loop for Continuous Improvement:** A loop of feedback where direct influence on the model training will be a very critical component for future development will exist. For instance, the users will mark emails that were wrongly marked as spam or not spam. Feedback will therefore exist in forms of machine learning models that feed back into the models learning from user preference and real world use cases. The machine will learn user's feedback, hence it will be an enhanced version of the model over time. That means it will be more personalized; yet more accurate. For example, to improve precision and recall metrics by better

understanding user-specific email patterns, this approach would help the system get to higher outcomes.

- **Advanced Machine Learning Techniques:** Future generation designs will likely include even more advanced models that involve machine learning, such as GANs or unsupervised learning algorithms to detect new forms of spam. Hybrid models that integrate benefits from multiple learning algorithms-have much promise to improve detection rates, particularly for complex or innovative spam methods.
- **Scalability and Cost Optimization:** Further, optimization of the system will be on the performance of cloud platforms by implementing more advanced autoscaling and cost control techniques. Through the use of serverless computing and pay-as-you-go models - two key cloud-native features - the system will ensure high availability and performance even at peak email traffic. Further, future work will be to minimize operational costs to further optimize resource utilization through strategies like Infrastructure-as-Code (IaC) and automated resource scaling.

The proposed enhancements will ensure that with the scaling up, such a system shall be effective, adaptive, and cost-efficient. And with factors related to continuous learning, integration of real-time intelligence, and user-driven customization, it shall be amply suited to face the reality of an ever-changing world of email-based threats.

8 Conclusion

The proposed system ensures the solution of spam detection by incorporating the available resources of cloud computing with advanced algorithms of machine learning. This proposed system overcomes the scalability and real-time performance challenges through the dynamic abilities of cloud platforms like AWS using powerful machine learning models, such as Logistic Regression, Support Vector Machines (SVM), Random Forest, Long Short-Term Memory (LSTM) networks, and Transformer models.

The evaluation, therefore, concluded with a marked improvement over traditional rule-based systems and standard ML methods in terms of precision, scalability, and security in spam detection. The system is robust, meaning it is strong, and applicable in the real world because it can handle big data, and process emails in real time and learns from the changing tactics of spammers. Further cloud-based infrastructure, therefore, ensures flexibility as well as cost-effectiveness and makes this solution scalable to different workloads and adaptable to various requirements of an organization.

It addresses the lacuna existing in the available literature to contribute to effective and adaptive spamming detection mechanisms. Future improvements will come in the form of advanced threats that are supported by intelligence, increased customization on filtering rules, and the implementation of user feedback loops for iterative quality improvement. In this regard, the system shall continue to maintain its relevance and potency in fighting spamming as it continues to evolve to combat the continuously changing threats to digital communication security.

Acknowledgement. We express our gratitude to the Rashtriya Raksha University, Gujarat, India, for providing necessary resources support and infrastructure for conducting this research. Special

thanks here are in order to the faculty and staff of the School of Information Technology and Cyber Security for their invaluable support throughout this work.

We would further like to thank the public dataset creators and contributors, Enron Email Dataset, particularly the SpamAssassin Public Corpus, who have been very instrumental in the training and testing of this research. It has been very important for validation purposes relating to the performance of the proposed system.

Compliance with Ethical Standards.

Conflict of Interest. The authors declare no contradictions of interest with the publication of this article.

Ethical Approval. This article does not contain human or animal studies conducted by the authors.

References

1. Goodman, J., et al.: Filtering junk E-Mail: a survey. IEEE Internet Comput. (2004)
2. Almeida, T.A., et al.: Spam filtering with the combination of classifiers. Int. J. Comput. Appl. (2011)
3. Zhang, Z., et al.: Convolutional neural networks for spam detection. IEEE Trans. Neural Netw. Learn. Syst. (2018)
4. Malhotra, P., Malik, S.: Spam email detection using machine learning and deep learning techniques. In: Proceedings of the International Conference on Innovative Computing & Communication (2022)
5. Armbrust, M., et al.: A View of Cloud Computing. Commun. ACM (2010)
6. Wu, Y., et al.: Cloud-based real-time spam detection. IEEE Trans. Cloud Comput. (2014)
7. Chang, V.: A Proposed Framework for Cloud Computing Adoption (2020). https://doi.org/10.4018/978-1-5225-9615-8.ch044
8. Tsai, C.-F., et al.: Combining machine learning and cloud computing for spam detection. J. Netw. Comput. Appl. (2019)
9. Li, R., et al.: A Hybrid approach to spam detection using cloud computing and machine learning. Cybersecur. Privacy J. (2020)
10. Sultana, T.: Email-based spam detection. Int. J. Eng. Res. (2020)
11. Kumar, N., et al.: Email spam detection using machine learning algorithms. In: Proceedings of the International Conference on Innovative Computing & Communication (2020)
12. Wolf, T., et al.: Transformers: State-of-the-Art Natural Language Processing (2020). arXiv:1910.03771
13. Jain, N., Sharma, B.K.: Email spam detection using machine learning. Int. J. Mech. Eng. (2022)
14. Ali, M.A., Lokhande, S.: Detect and analysis of spam email based on machine learning algorithms – comparative approach. J. Xi'an Shiyou Univ. Natl. Sci. Ed. (2022)
15. Amazon Web Services. An Overview of the AWS Cloud Adoption Framework (2022). https://docs.aws.amazon.com/pdfs/whitepapers/latest/overview-aws-cloud-adoption-framework/overview-aws-cloud-adoption-framework.pdf
16. Uzoma, B., Okhuoya, B.: A research on cloud computing. Int. J. Commun. Netw. Syst. Sci. (2022)

17. Islam, R., et al.: The future of cloud computing: benefits and challenges. Int. J. Commun. Netw. Syst. Sci. (2023)
18. Sambasivarao, L., Kumar, K., Chandra, P., Naga, N., Galla, I.: AWS and future of cloud computing. Int. J. Adv. Res. Comput. Commun. Eng. (2024)
19. Edricgsh. How To Design a Spam Filtering System with Machine Learning Algorithm. Towards Data Science (2018). https://towardsdatascience.com/email-spam-detection-1–2-b0e06a5c0472
20. Tabish, A.: Machine Learning Techniques for Spam Detection in Email (2022). https://medium.com/@alinatabish/machine-learning-techniques-for-spam-detection-in-email-7db87eb11bc2
21. UCI Machine Learning Repository. SMS Spam Collection Dataset (2016). https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset/data
22. Olalekan, G.: Spam Assassin Email Classification Dataset (2021). https://www.kaggle.com/datasets/ganiyuolalekan/spam-assassin-email-classification-dataset



From Pen to Pixel: Understanding Learning Shifts with Student Insights in E-Education

Kavita Yagyanarayan Jhanwar(✉) and Priti Sadaria

Department of C.S. & I.T., Atmiya University Rajkot, Rajkot, India
krishnaradha252525@gmail.com

Abstract. The fast development of information technology has greatly altered many parts of our lives, especially in the areas of employment, social relationships, and schooling. By bringing students into the classroom and encouraging active participation from them, online education is changing the way people learn and is a key driver of the current digital revolution. Examining how asynchronous and synchronous learning platforms might enhance efficiency in comparison to traditional methods, this study delves into the dynamic landscape of e-education. Online chats, video conferencing, and other types of online connection are all a part of these platforms. Education has become more accessible and flexible due to the expansion of online and blended learning. This trend was made possible by the widespread use of the Internet in the 1990s. Benefits of online education, such as reduced reliance on physical infrastructure and cost reductions, are being bolstered by the ever-increasing sophistication of information technology. This study delves into the impact of online education on promoting continuous learning and career advancement during the COVID-19 pandemic. It zeroes focused on the possibilities of digital platforms for lifelong learning, reskilling, and skill acquisition. The growing area of digital learning benefits from the study's insightful and well-reasoned opinions.

Keywords: Evolution · E-learning · Education · Training · Trends · Learning Management Systems · Platforms

1 Introduction

The extensive influence of information technology has significantly affected several aspects of our lives, including employment, social connections, data manipulation, information evaluation and dissemination, amusement, and travel encounters. The e-evolution, also known as e-revolution, has witnessed the rise of various digital advancements including e-mails, e-commerce, e-government, and presently, e-education. Online education, also referred to as e-education, is fundamentally transforming the process of acquiring knowledge. Participate in the process of teaching and gaining information. The progression of education delivery models has been rapid and transformative. The worldwide adjustment of institutions to these advancements has resulted in a highly dynamic education landscape, generating substantial attention from scholars, educators,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 326–338, 2025.
https://doi.org/10.1007/978-3-031-86296-0_23

administrators, policymakers, publishers, and companies. Traditional "correspondence" lessons began in England around the middle of the 19th century. E-education, on the other hand, offers faster and more efficient ways to teach. Users can access online discussion boards, chat rooms, and videoconferencing through this tool, which also lets users learn at the same time or at different times. The Internet and World Wide Web came out in the 1990s. This was the start of modern "online" or "blended" learning. This type of schooling is designed for individuals residing in remote places or those who wish to avoid the necessity of traveling. Online education has grown more feasible in terms of technology, economics, and operations due to the ongoing advancements in information and communication technologies. Universities are compelled to provide online programs as a result of cost constraints and advantages, such as reduced reliance on physical infrastructure such as classrooms, offices, cafeterias, dormitories, and libraries. Moreover, the accessibility of sophisticated technology facilitates the seamless integration of online programs there is a growing presence of nontraditional students who are employed full-time and have a preference for online learning. Online education in the United States has experienced significant growth, which can be observed through four distinct levels. In addition, the lines following a table, figure, equation, or similar elements lack a gap.

2 Evolution of Online Education

The term "distance learning" has evolved into "online learning" during the past three centuries. The concept originated with correspondence courses, which allowed students to learn remotely through mail. Later developments involved the incorporation of mass communication mediums, particularly radio and television, which expanded the availability of education. Online learning has achieved unparalleled success in the modern digital era, utilizing user-generated content and interactive platforms. This evolution not only demonstrates the adaptability of education but also emphasizes the ongoing interaction between learning and the changing technology environment, promoting a varied and inclusive educational experience [1]. Similar to today's instructors, it has to be claimed that the quality of instruction would be equivalent to that of traditional inperson instruction. A professor from the University of Wisconsin started the first nationally authorized radio station in 1919. This was the first time that radio and TV were used as teaching tools. As early as the 1950s, colleges teamed up with TV stations to offer classes that could be taken for school credit. The University of Phoenix is known for having a wide range of classes that can only be taken online. It started using online technology in 1989 through CompuServe, which was one of the first online service providers, and then switched to the World Wide Web in 1991. New York University was one of the first schools to offer an online program when it started NYU Online in 1998. Despite the fact that many other online programs came out later, most of them did not last. In 1999 and 2000, "blended" or "hybrid" programs came about because people were unhappy with online programs. In these schools, both in-person and online classes were used together to get the most out of both. New technologies allow both asynchronous and **synchronous delivery** choices for online and blended programs. Online talk rooms, discussion boards, and videoconferencing are all used in these programs [2].

3 Types of Digital Learning

The study highlights the significance of implementing digital learning in educational institutions and stresses the necessity of surpassing the mere presence of digital technology within the classroom. It emphasizes the importance of teachers' proficiency, namely in their fundamental digital skills and technology-oriented teaching abilities, in successfully incorporating digital technology into instruction and fostering exemplary teaching with technology. Given the COVID-19 pandemic and the growing demand for remote teaching, this study highlights the significance of incorporating technology-related teaching skills into both initial teacher education and ongoing professional development. This will enhance the chances of students engaging in productive learning activities using digital technology [22]. How does digital learning impact education? That is the central question explored in this special edition of Digital Education Review. Among the numerous advantages of digital learning are the following: the development of students' employability skills, the simplification of access to educational materials, and the expansion of opportunities for individualized instruction. Furthermore, it has the potential to engage and inspire students, thereby equipping them to face the challenges of a technologically sophisticated world [23]. Multimedia resources, interactive simulations, and online discussion forums are just a few of the learning tools that students have access to in online classrooms. They can also tailor lessons to the needs of individual students, allowing them to study at their own speed while receiving constructive criticism. More students and teachers may be able to communicate and collaborate in digital learning environments, which can increase the amount of time spent on group projects. For students to be motivated and successful in higher education, it is necessary for them to feel like they belong and are actively involved [24]. Figure 1 shows a quick summary of the most recent changes in education that have come about because of the progress of digital technologies. The term "e-learning" refers to a way of learning that uses computers and the internet. M-learning is a type of digital learning that came about because of mobile phones and portable internet. Sensor technology, as well as mobile and wireless transmission technologies, are needed for the idea of ubiquitous learning to work. The dynamic role of technology in education poses both obstacles and prospects, significantly influencing the future of learning. To gain a thorough comprehension of these patterns and their influence on learning inclinations in the field of education.

Figure 1 indicates the mobility and integration of three distinct types of digital learning—E-learning, M-learning, and U-learning—aimed at comparing their adoption within educational institutions. E-learning is predominantly stationary, necessitating the use of desktop computers or laptops, characterized by a high degree of integration but limited mobility. In contrast, M-learning, which depends on portable devices such as smartphones, offers considerable mobility but demonstrates a lower level of embedded integration. U-learning, or ubiquitous learning, is designed to adjust to the learners' environments, providing fluid and customizable educational experiences across various contexts, marked by high mobility and extensive integration.

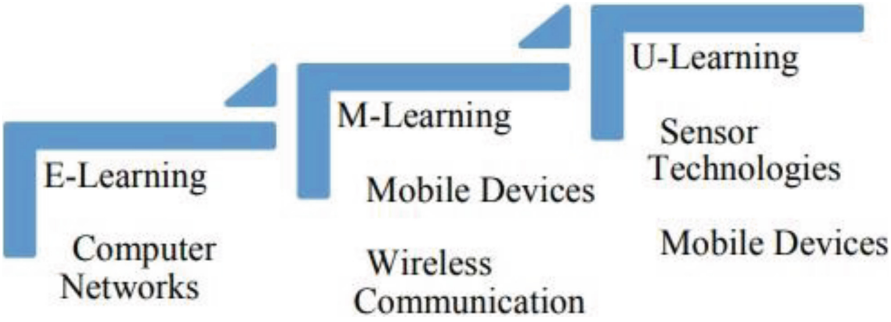


Fig. 1. A comparative analysis of the effectiveness of E-learning, M-learning, and U-learning environments in enhancing educational experiences.

3.1 E-Learning

Since the implementation of E-Learning, the educational environment has witnessed a significant transformation. It has revolutionized how knowledge is acquired and shared in the society and other related areas of interest. Hence, E-Learning is significantly cheaper than traditional teaching practices owing to the ability to cut down on teacher training as well as the usage of paper and pencil. The subject gains advantages when learning becomes flexible as it allows the students to be involved in the learning process at their own convenience regardless of their stationery or time. As opposed to normal teaching strategies, E-Learning allows the students to follow the necessary steps on their own because the training material is not limited by the trainer or by any organization. Students' attitude towards learning and academic achievements are enhanced by the online education as revealed in various studies. It has opened up the education regime for individuals who were hitherto restricted by finance or physical distance. Altogether, the utilization of E-Learning has enhanced the delivery of education and created new avenues for justice for both faculty and students [24]. A number of studies have been conducted to; analyses the possible effects of online education The risks involved have being debate widely within networking academic circles. There is variety of benefits associated with online classes such as flexibility, cost-effectiveness, availability, and individualization. E-learning comes in as a package that enables the learners gain control over their preferred time and place of study since the courses are set at flexible paces. This enhances the intake of higher learning institutions for those with difficulties of finance or distance to cover.

Computer Networks. The adoption of computer networks in e-learning has really changed the face of education. For students and educators there are now numerous opportunities and benefits which can be listed. Synchronous and asynchronous computer mediated communication tools like CWBs, BBSs DBs, and e-mail are efficient tools when it comes to learning intervention. Computer based instruction can increase a student's independence since learners are able to explore the learning networks independently and express themselves in the process. The students can communicate with many people of different cultures, exchange information and documents, collaborate

on the writing assignments, that enables them to study independently. Computer networks allow the general access to education materials like multimedia and lectures from wherever one is. Besides, these plat-forms help in engaging in communication where the student is able to post messages, comment, and even work on projects with his/her teachers and classmates. Computer networks bring to students the benefits which consist in availability of various educational resources: libraries, journals, and study databases which can be reached through the Net. That in turn enriches the education opportunities for pupils and offers them what is not provided by their current curricula. Some of the problems that should be taken into consideration in the case of studying with the help of computer networks include the student's inexperience and inability to work with computer networks or inadequate access to the resources needed for their lessons, and their discomfort or unfamiliarity with the forms of remote education. The continuing education design requires collaboration with programme designers, instructors, and technology specialists to solve the multifaceted issues. Therefore, the assignment of student's IT support through computer networks is vital for the proper functioning of e-learning activities. Thus, it is imperative to understand and meet the technical and instructional needs of the students concerning computers and computer networks for successful implementation of e-learning.

3.2 M-Learning

M-learning provides numerous advantages in educational settings. It enables users to access content at any time and from any location, facilitates remote learning, improves student-centered learning, allows for flexible just-in-time training or content review, and is beneficial for those with disabilities. M-learning facilitates the customization of student learning based on individual needs and preferences, fosters engagement among students, learners, and instructors, and mitigates cultural and communication obstacles between faculty and students. In general, mobile learning (M-learning) offers easily accessible and widely available learning options, fundamentally transforming the conventional education system by offering greater flexibility, tailored instruction, and interactive learning experiences for both students and educators. The primary advantages of mobile learning (M-learning) in educational settings encompass cost-effectiveness, the ability to deliver and create multimedia content, ongoing and context-specific learning support, reduced training expenses, the potential for a more satisfying learning experience, and improvements in literacy, numeracy, and participation.

Mobile Devices. The utilization of mobile devices in E-Learning, also referred to as M-Learning, has experienced a significant rise in popularity as a result of the widespread existence of mobile devices. With mobile e-learning, students can use their own phones to access course materials online from anywhere at any time. Some of the ways that mobile devices let you enter information are through touch screens, QWERTY keyboards, accelerometers, and GPS sensors. Because of these traits, they are better able to change in the classroom. Many tools and ways of teaching have become easier to get to through mobile devices in E-Learning. For example, games can be used to teach course material. With this way, learning is better because it makes things more fun and easier. On the go, users can access games that teach them new things while they go about their

day. Mobile learning technology that encourages engaging and collaborative learning has made it possible for students to share what they know and get to their learning materials whenever and wherever they want. People who can't go to more traditional schools because of health issues, money issues, or other reasons can really benefit from this ability to change. Students can make their own lessons with mobile learning because it gives them a lot of freedom to work around their needs and hobbies. Adaptive learning systems can see how well students are doing by looking at data and changing lessons as needed.

Wireless Communication. Wireless communication technology can be employed in e- learning to augment the learning experience for both students and educators. Wireless networks enable students and instructors to access online resources and communicate with one other from any location, hence providing internet connectivity. This can be especially advantageous for distance learning programs, as it allows students who live across the country to participate. Wireless communication technology have been crucial in facilitating e-learning during the COVID-19 pandemic. Wireless networks have enabled remote education by offering connectivity for students and professors. Moreover, the reliance on wireless communication technology amid the epidemic has opened up the possibility of a future transition towards a remote employment and a hybrid schooling system. It is crucial to acknowledge that although wireless communication technologies have facilitated e-learning, obstacles such as the digital divide and the requirement for dependable, high-speed internet connections persist. It enables the efficient transmission of educational material, cooperation, and instantaneous communication. Wireless communication in e-learning provides numerous advantages, such as flexibility, accessibility, and the capacity to involve learners in various situations. It facilitates the incorporation of multimedia components and interactive educational encounters.

3.3 U-Learning

U-Learning, or Ubiquitous Learning, is an educational method that utilizes contemporary technology to facilitate uninterrupted learning at any time and in any location. It is distinguished by the incorporation of information technology and unrestricted communication, enabling universal access to learning. U-Learning utilizes computer and internet technology to enhance learning in several settings, such as formal schooling, professional growth, and lifetime learning. This method prioritizes the smooth incorporation of learning into daily activities, enabling learners to interact with educational content and resources in various settings. The objective of U-Learning instructional models, is to augment students' capacity for creative and innovative thinking and enhance the overall learning experience in higher education environments. The key features of ubiquitous learning include durability, availability, promptness, interactivity, contextualization of instructional activities, responsiveness to the learner's present circumstances, and non-obtrusiveness. This highlights the transition from conventional educational frameworks to accommodate u-learning, which occurs at any location and time, and across various learning settings.

Sensor Technologies. Sensing and responding to physical signals like motion, sound, light, and temperature are the capabilities of sensor technologies. Through the use of

these technologies, online education can improve the quality of instruction while also giving students access to more personalized and hands-on approaches to knowledge acquisition. As an example, sensors can be applied to monitor how active and healthy a student is when s/he is taking an online lesson. Also, teachers and instructors may use them to find out how engaged and attentive their students are during their lessons; which is pretty valuable information as to how they may improve the lessons in terms of entertainment value. Sensors can also be used to generate other virtual reality models that make up some of the forms of learning environments. These assist the students to explain difficult lessons through touch and in groups. All in all, speaking of the current and upcoming sensor technologies, one can presume that online education, in general, can be changed significantly with the help of timely and interest-focused interactions.

Mobile Devices. Mobile gadgets encompass devices such as cell phones and other compact, lightweight technological tools. This category encompasses all mobile phones, tablets, and computers. These tools have transformed our ability to communicate and access information, making them indispensable in our daily lives. Mobile devices change e-learning by providing the capability to access learning materials and information while on the move. Consequently, students are empowered to exert greater autonomy over their education, enabling them to learn at their preferred speed and in their preferred settings. Mobile devices can also serve as a means to engage in online discussions, cooperate with classmates, and turn in assignments. Mobile devices have numerous benefits in e-learning, such as easy access, increased involvement, customization, and availability of resources. Learners are granted the ability to retrieve instructional materials and resources at any given time and location, so offering them flexibility and convenience in the learning process. The interactive functionalities of mobile devices, such as touchscreens and multimedia capabilities, can augment student engagement and involvement in e-learning activities.

Figure 2 Shows a comparison of learning settings based on their levels of embeddedness (low to high) and mobility (low to high). In this case, mobility—how easy it is to move around—and embeddedness—how well they work together—are features that highlight how much memory and storage space is used. As a result, we can see that computer- assisted learning is not as good as either of them. For example, e-learning is not very mobile but much embedded, while m-learning is very mobile but not very embedded. On the other hand, u-learning has a high amount of embeddedness, which means it can store a lot of information and is easy to move around.

This Figure delineates four distinct educational methodologies: Computer-assisted Learning, E-learning, M-learning, and U-learning, each showcasing varying levels of mobility and integration within educational structures. Computer-assisted learning is marked by limited mobility and integration, as it depends on stationary technology. E- learning offers enhanced integration within educational frameworks, though it still faces challenges related to mobility. M-learning enhances mobility through the use of portable devices such as smartphones; however, its application in formal educational settings remains somewhat limited. In contrast, U-learning is distinguished by its significant mobility and profound integration, facilitating flexible, context-aware learning experiences that adapt to the learner's environment at any time and location. This underscores the superior adaptability of U-learning in contemporary education.

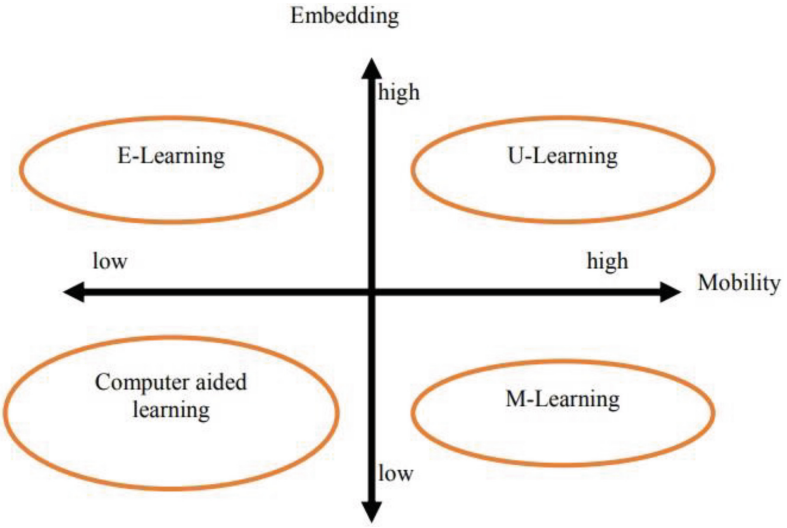


Fig. 2. Comparison of the mobility and embeddedness of Computer-assisted Learning, E-learning, M-learning, and U-learning environments.

4 Methodology

The methodology involved collecting data from participants through a Google Forms survey to obtain quantitative information. The survey was designed to explore students' perceptions, emotions, and challenges related to online education. Multiple-choice questions using a Likert scale were used to capture different perspectives. Google Forms was chosen due to its user-friendly interface, ability to reach a wide range of users, and ability to automatically categorize responses for improved processing and analysis. This approach ensured complete coverage and reliable data, suitable for statistical analysis and complementary to qualitative results.

5 Types of Digital Student Perspectives on E-Learning Environments

The main results of the research showed that e-learning gave students more freedom in choosing between instructor-led or self-study classes, so they could learn anywhere and at any time. That being said, one of the problems that was pointed out was that e-learning made it harder for friends to meet up in person. Distance learning, also known as e- learning, is very popular at universities and other schools. There are more part-time, non-residential, and non-traditional students, which has increased the need for distance learning choices. There are, however, worries about how well e-learning works and whether or not end users will accept it as a way to learn. The main complaint against e-learning is that students feel alone because they can't talk to each other face-to-face. A lot of students and teachers think that e-learning is cold and impersonal, and that it takes away the important chance for students and teachers to talk openly about what they are learning in class (Table 1).

Table 1. Survey results showing student perceptions of the benefits and challenges of e- learning environments.

Sr. No	Question	Strongly agree	Agree	Disagree	Indifferent
1	The e-learning environment empowers us to access information and learn at our convenience.	27.5%	68.6%	0%	3.9%
2	E-learning helps me study more effectively than traditional learning.	22.5%	56.9%	7.8%	12.7%
3	A good environment can make it easier for teachers and students to communicate and share information.	32.4%	58.8%	3.9%	4.9%
4	E-learning needs technical skills.	32.4%	60.8%	2%	4.9%
5	I believe that the e-learning environment can boost	22.5%	59.8%	7.8%	9.8%
6	The e-learning is only advisable for people with a lot of computer knowledge.	23.5%	45.1%	24.5%	6.9%
7	Online learning uses a lot of time and stresses me out a lot.	17.6%	52.9%	16.7%	12.7%
8	There will be more helpful resources in the e-learning setup.	21.6%	70.6%	3.9%	3.9%
9	I find it hard to become good at using the e-learning platform.	20.6%	52%	20.6%	6.9%
10	I am not happy with how the e-learning allows me to interact with other students.	13.7%	49%	25.5%	11.8%

This Tables highlights An analysis of 102 responses reveals the diverse perspectives of students toward online education. The consensus is that technology is both convenient and efficient. Specifically, 68.6% of individuals believe that it enables easy access to knowledge, while 56.9% assert that it surpasses traditional methods in terms of educational effectiveness. Additionally, 58.8% of respondents recognize that technology has the capacity to enhance communication between teachers and students. However, a significant minority, comprising 60.8% of the overall population, consider technological competence to be essential for online education, indicating that it could pose a challenge for certain students. An enhancement in the user experience and support techniques for the e-learning platform is necessary, as 52% of individuals report difficulties in comprehending it. Notwithstanding these challenges, 70.6% of participants express optimism over the ample availability of resources provided by e-learning. However, 25% of students express dissatisfaction with the level of engagement they may attain using online learning platforms. To optimize online education, these studies emphasize the importance of eliminating technological barriers, enhancing platform usability, and fostering student engagement in their own learning (Table 2).

Table 2. Survey results showing students' perceptions of the benefits, challenges, and overall satisfaction with e-learning environments.

Factor	Computer literacy and skill	Computer availability and accessibility	Platform Function	Training	Instructor attitude	Perceived ease of use	E-learning course quality	E-learning platform flexibility
Students (%)	11.8%	25.5 %	3.9%	24.5%	21.6%	0%	12.7%	11.8%

This table summarizes the results of a survey focused on students' views of the e-learning environment. The key findings indicate a strong consensus on the accessibility and flexibility offered by e-learning, with 68.6% of respondents stating that it enables them to learn at their convenience, and 56.9% believing it to be more effective than traditional educational methods. However, challenges related to technology persist, as 60.8% of students reported that e-learning requires a considerable level of technical skill. Notably, one-quarter of the students expressed dissatisfaction with the limited opportunities for interaction with peers. These findings underscore the necessity for enhanced technical support and a more interactive learning atmosphere to boost student satisfaction (Fig. 3).

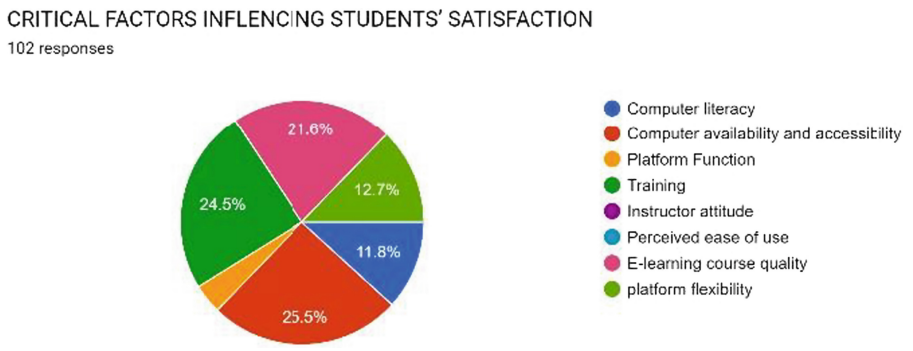


Fig. 3. Factors affecting student satisfaction in online education based on accessibility, technology use, and instructor support.

Analysis of survey responses from 102 students reveals important insights into the factors influencing their confidence in online learning. Despite the fact that most respondents (97.97%) are confident in their ability to understand complex scientific material and perform well on exams and assignments (94.94%), the survey does provide some insight into the sources of this confidence. The accessibility and availability of computers was deemed very influential by 25.5% of the pupils when many factors were taken into consideration. The instructor's demeanor and the training they receive both have a significant impact on students' confidence levels (24.5% and 21.6%, respectively). On the other hand, factors like platform adaptability, perceived user-friendliness, quality of

e-learning courses, and computer knowledge and competency have less of an impact. This suggests that while these elements do affect the quality of the e- learning process as a whole, they might not be as important in determining the level of self-assurance that students feel. These data evidence the necessity of self-efficacy to enhance students' motivation and increase their performance within e-learning environment. To develop self-efficacy, the effective teaching methods for learning have to be employed and there should be supportive environment around. Extending the students' confidence and helping them to perform better academics within online classes require improving computer access, offering extensive practice, and developing sound rapport between teachers and learners. The Composite E-Learning Attitude Score or the CEAS was therefore developed in a bid to gather a variety of important attributes that define e-learning into one score. The CEAS which is made up of, AK for Access to knowledge, EE for Educational effectiveness, CE for Communication enhancement, TCC for Technological competence, PUC for platform usability, RA for Resource availability and last but not the least SE for Student engagement. The importance of each component with respect to other components is represented by the weight, which belongs to a component. The weights were assigned as follows: 0.2 for AK, 0.2 for EE, 0.15 for CE, –0.1 for TCC, –0.1 for PUC, 0.15 for RA, and 0.1 for SE. The resulting formula is

$$CEAS=(w_{ak}\times AK)+(w_{ee}\times EE)+(w_{ce}\times CE)-(w_{tcc}\times TCC)-(w_{puc}\times PUC)+(w_{ra}\times RA)+(w_{se}\times SE)$$

$$CEAS=(0.2\times 68.6)+(0.2\times 56.9)+(0.15\times 58.8)-(0.1\times 60.8)-(0.1\times 52)+(0.15\times 70.6)+(0.1\times 75).$$

The mean score for e-learning attitudes, calculated by summing all the values, is 40.73. Based on this data, this is the prevailing sentiment among the majority of students to- wards e-learning. The advantages of technology lie in its facilitation of access to knowledge, tools, and fostering curiosity. However, certain issues arise from a lack of proficiency in utilizing technology and a lack of familiarity with the tool. This substantial figure underscores the significance of rectifying technical issues and enhancing user-friendliness in order to fully harness the advantages of e-learning.

6 Challenge

The rapid progression of e-learning technology has introduced significant technological problems that impact the effectiveness of online education. A primary concern is the digital gap, since disparities in access to technology and reliable internet connections hinder fair engagement among pupils. Studies indicate that students from low-income households frequently encounter obstacles stemming from insufficient resources, hence constraining their capacity to effectively engage with online learning platforms. To address this issue, it is crucial to develop new solutions, like the provision of subsidized gadgets and the enhancement of internet access in underprivileged regions. Another technological challenge is guaranteeing the efficacy and accessibility of online learning platforms. Numerous contemporary systems exhibit inadequate usability, complicating navigating for users, particularly those with impairments. Research indicates that enhancing user experience through optimal interface design can result in increased involvement and satisfaction levels. This necessitates ongoing user feedback to guide design enhancements

and ensure that the platforms meet the diverse demands of students. Delivering technical training to students and educators is crucial for the successful execution of online learning. A significant number of educators lack the proficiency to proficiently use technology into their pedagogical approaches. Research underscores the need of extensive professional development programs emphasizing digital literacy, pedagogical techniques, and the proficient application of educational resources. An effective technological support system is necessary. Students frequently have technological difficulties that may impede their academic advancement. Establishing accessible and responsive support services may significantly enhance the e-learning experience by promptly addressing user concerns. Addressing these concerns will promote the establishment of a more inclusive and successful online learning environment capable of accommodating the unique needs of all learners.

7 Conclusion

Due of its increased flexibility, accessibility, and customizable features, e-learning has become an essential tool in higher education. Despite its drawbacks of limited practical experience and less engagement, e-learning is still a preferred option for many schools due to its advantages. Additional study is required to investigate the efficacy of e- learning in various settings and for diverse learner profiles as the e-learning domain progresses. Investigation is required to examine the most effective methods for creating and implementing e-learning courses, as well as the influence of e-learning on faculty training and improvement. Customizing mobile learning environments for individual users, integrating diverse resources and learning methods, and employing games to deliver course information enhance the engagement and accessibility of learning. Additional research in fields such as user experience and engagement, adaptive learning technologies, mobile learning analytics, accessibility and inclusivity, integration with emerging technologies, and cross-cultural studies can contribute to the progress of mobile E-Learning, ultimately improving its effectiveness, accessibility, and influence on education.

References

1. Three Generations of Distance Education Pedagogy Anderson and Dron
2. Vaiciukevičiūtė, A., Stankevičienė, J., Bratčikovienė, N.: Higher education institutions' impact on the economy. J. Bus. Econ. Manag. **20**(3), 507–525 (2019). https://doi.org/10.3846/jbem.2019.10156
3. Firmansyah, R., Putri, D.M., Wicaksono, M.G.S., Putri, S.F., Widianto, A.A., Palil, M.R.: Educational transformation: an evaluation of online learning due to COVID-19. Int. J. Emerg. Technol. Learn. **16**(7), 61–76 (2021). https://doi.org/10.3991/ijet.v16i07.21201
4. Faize, F.A., Nawaz, M.: Evaluation and improvement of students' satisfaction in online learning during COVID-19. Open Praxis **12**(4), 495 (2020). https://doi.org/10.5944/openpraxis.12.4.1153
5. Impact of coronavirus pandemic on education. J. Educ. Pract. (2020). https://doi.org/10.7176/jep/11-13-12
6. IMPACT OF PANDEMIC COVID (2020). https://doi.org/10.24941/ijcr.39209.07.2020

7. Sarica, G.N., Cavus, N.: New trends in 21st century english learning. Procedia Soc. Behav. Sci. **1**(1), 439–445 (2009). https://doi.org/10.1016/j.sbspro.2009.01.079
8. Yang, H.H.: New world, new learning: trends and Issues of E-Learning. Procedia Soc. Behav. Sci. **77**, 429–442 (2013). https://doi.org/10.1016/j.sbspro.2013.03.098
9. Molebash, P.: Technology and Education : Current and Future Trends
10. Bali, S., Liu, M.C.: Students' perceptions toward online learning and face-to-face learning courses. In: Journal of Physics: Conference Series, Institute of Physics Publishing (2018). https://doi.org/10.1088/1742-6596/1108/1/012094
11. Hirapara, J., Vanjara, P.: A comparative study of data mining techniques for agriculture crop price prediction. In: 2022 IEEE 7th International conference for Convergence in Technology (I2CT), pp. 1−6. Mumbai, India (2022). https://doi.org/10.1109/I2CT54291.2022.9824533
12. Castle, S.R., Mcguire, C.J.: An Analysis of Student Self-Assessment of Online, Blended, and Face-to-Face Learning Environments: Implications for Sustainable Education Delivery (2010). www.ccsenet.org/ies
13. Pearson, J.: Investigating ICT using problem-based learning in face-to-face and online learning environments. Comput. Educ. **47**(1), 56–73 (2006). https://doi.org/10.1016/j.compedu.2004.09.001
14. Markova, T., Glazkova, I., Zaborova, E.: Quality issues of online distance learning. Procedia Soc. Behav. Sci. **237**, 685–691 (2017). https://doi.org/10.1016/j.sbspro.2017.02.043
15. Solanki, M., Gujarati, P.: The digital revolution in recruitment: unraveling the impact and challenges of e-recruitment. Educ. Adm. Theory Pract. **30** (2024). https://doi.org/10.53555/kuey.v30i6(S).5362
16. Bates, A.W.: Permission To Reproduce This Material Has Been Granted By
17. Doshi, M., Hirapara, J.: A study on data mining techniques for forecasting FMCG product sales. In: 2024 15th International Conference on Computing Communication and Net- working Technologies (ICCCNT), pp. 1–6. Kamand, India, (2024). https://doi.org/10.1109/ICCCNT61001.2024.10726077O.
18. Jethro, O.O., Grace, A.M., Thomas, A.K.: E-Learning and Its Effects on Teaching and Learning in a Global Age (2012). www.hrmars.com/journals
19. Amit, K., Patel, A., Atkotiya, K.H.: A review paper on seo for ranking and effectiveness techniques in context of google search engine, paripex-indian. J. Res. **9**(1) (2020)
20. Ally, M., Prieto-Blázquez, J.: ¿Cuál es el futuro del aprendizaje móvil en la educación? RUSC Univ. Knowl. Soc. J. **11**(1), 142–151 (2014). https://doi.org/10.7238/rusc.v11i1.2033
21. Vachhani, P., Parsana, F.: A comparative study of a hybrid recommender system merging collaborative and content-based strategies for nutritional tailoring in children. In: 2024 First International Conference on Pioneering Developments in Computer Science & Digital Technologies (IC2SDT), pp. 381–386. IEEE (2024). https://doi.org/10.1109/IC2SDT62152.2024.10696128
22. Romani, C.C., Rivera-Vargas, P., Andrés Bello, U., Cristóbal Cobo Romani, C., Bank, W.: Editorial: Digital learning: distraction or default for the future. http://greav.ub.edu/der/
23. Doshi, M.: A comparison of data mining approaches for forecasting sales of FMCG food products. In: 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1–7. Delhi, India (2023). https://doi.org/10.1109/ICCCNT56998.2023.10307432
24. Warschauer, M., Turbee, L., Roberts, B.: Computer Learning Networks And Student Empowerment (1996)

Navigating the Path: Exploring Career Choice Dynamics Among Secondary and Higher Secondary School Students

Falguni Mukeshbhai Barad(✉) and Amit Khjimjibhai Patel

Department of Computer Science, Atmiya University, Rajkot, Gujarat, India
falgunibarad09@gmail.com, amit.patel@atmiyauni.ac.in

Abstract. With an emphasis on individual jobs and the limitations that limit their choices, this research study aims to investigate the factors that influence career training choices made by students in secondary and higher secondary schools. Students' ability to make good job decisions depends on their career decision self-efficacy, which is impacted by the success they've had in exploring other career paths and the presence of supporting surroundings. Using a sequential research approach, the study looked at how factors like gender inequality, educational chances, social pressure, parental pressure, and personal interest influenced each other. Successful career development programs and related educational endeavors can be informed by the findings. The findings also back strong arguments about how school, peer pressure, and parental obligations affect children's career choices. Student anxiety about choosing poor career choices stems from five main places: a lack of self-awareness, inadequate investigative abilities, cultural and familial value adherence, emotional difficulties, and the fear of failure. An integrated approach utilizing various elements, such as encouraging personal growth, developing curiosity, following parental and teacher direction, making sure appropriate resources are available, and putting an emphasis on practice, is necessary to effectively address these problems.

Keywords: Career selection · higher secondary students · exploration · gender differences · challenges · career counselling

1 Introduction

There is need to establish the relationship between the careers that students choose in secondary and higher secondary schools and their characteristics or else the available support for the students. Self-efficacy in career decisions is highly correlated with the creation of goals for career exploration and environmental support such as professional information that has been realized to have an impact on the students' decision-making process. Therefore, encouraging students into the activities that enhances the career and that which is focuses on building inner criterion of the student may help in improving the internal psychological aspects that are imperative in decision making at the client. Professions which are very central in decision-making processes particularly those affecting

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 339–349, 2025.
https://doi.org/10.1007/978-3-031-86296-0_24

the lives of people of color. Using the sources listed online and the information introduced to the students, watching job videos, and thinking about one's values and desires, all helps the students in their process and expands the vision on the world of professions that matches their skills and interests. This paper shows, how, based on possibilities to gain insights into careers of their choice, opportunities for reflection on, and planning of their future, educators' and stakeholders' can prepare students for the career spheres. In addition, the tools such as career interest tests, work importance personality surveys, online career resources, career magazines, mood boards and statistics of careers will assist students in their journey of career decision making. Explores the challenges introduced by learners in school as well as in the learning institutions in matters concerning choice of career. Concerning its title and purpose, the article implies identifying the factors that shape learners and their choices regarding the career they aspire to. The objectives of the study will be to comprehend the psychological factors and sequences that influence the development of work-related aspirations of students. The objectives of the research are to determine the factors that affect the students' choices comprising of the subjects and career options, the role of family in students' choices, the pressure that the society puts on students and the available educational opportunities for the students. The process is complex and affected by various internal and external factors. The comprehensive understanding of the issues at hand is achieved through the acquisition of qualitative insights derived from closed-ended questions and in-depth interviews. The goal of this study is to examine the impact of several factors, including gender, media, academic motivations, family, peers, teachers, and personal interests, on students' career choices.

2 Factors Influencing Career Choice

The factors that influence a person's choice of occupation are multifaceted and may vary from one individual to the next. The file contains the study's conclusions, which highlight many important elements. Because many jobs call for certain character traits, one's personality plays a significant role in the job search. An outgoing personality, for example, could be necessary for sales roles. The opportunities that people have at their disposal also have a role in their career decisions. The career paths that students choose to follow are greatly affected by the opportunities that are available to them, or the lack thereof. In order to make educated choices regarding one's career, it is essential to have a thorough awareness of one's own strengths, weaknesses, interests, and personality. The results underscore career identity exploration, with self-initiated processes being more critical during the early exploration stage. A person's performance in school, class record, and scholastic achievements could very largely determine his/her employment opportunities in the future. Education is a crucial factor for all learners, given that education determines employment opportunities in the market [1]. It is important to note that, there are several factors that may influence an individual to choose a particular occupation. According to the research findings, there are many facets that greatly affect students in making decision where to focus, or what to do in future. More often than not, parents play a major role in determining which direction their children should take when choosing a career. Moms have been observed to have more significance in the

children's lives more than dads, according to various scholars. Another essential factor is students' parents' education level, occupation and income. Both environmental as well as individual factors that encompass financial motives, personal desires, education type, economic status, parent example, friends' example, media influence and other sources combine to provide the professional decisions made by children. For educators and policymakers who are interested in supporting the youngsters in the best ways possible when it comes to career decisions, they should endeavor to have adequate knowledge of these important aspects [2].

2.1 Family Influence Factors

Among several factors that may influence their work decisions, children rely so much on their parents. Empirical evidence shows that maternal influence is preeminent compared to paternal influence. It is pertinent to point out that parents' income along with their professions and even education can work significant roles to set up children's future occupations. It can also be hypothesized that the level of support provided by the family and how close knit the family is might determine the likelihood of a child achieving his or her desired vocation in future. Therefore, based on the given comparison the parents' occupations can influence children in a way that only children ought to be supported and live in an environment that does not distract them. Therefore, according to the study, the family environment stimulates, motivate, encourage and therefore can influence a child's professional dreams [3]. However, using parents' education level, the investigation realized that there is no valid correlation between SES with students' performance as has been infield. This mean that the educational level of parents may not dictate the kind of jobs that children would pursue in the future. The conclusion of the said study indicated that parents have liberal attitudes in the aspect of career path where the children are on their own but assisted by the parents if needed. In most cases there is no clash between the parents and the children when it comes to choice of careers but sometimes, they may have some concern or they may have something to say either good or bad about the career one wants to choose. This study also pointed out that parents could shape one's career indirectly due to the educational level of siblings to conclude that they do not want to progress to college after scoring well in high school [4].

2.2 Peer Influence Factors

The pressure from the peers plays a big role in influencing a person in decision making, especially the one concerning career. Research shows that friends play a significant role when it comes to choosing a career in life. Individuals who were born in the Y age. Nonetheless, the kids insist that, even though it may not seem so, friends do influence the careers that they take. But this means that friends can assist by passing information or even in passing jobs but the decision to make certain choices can become more individual. According to the findings of the study, friends have the same impact on the career choice as teachers in the life of a person. This means that friends at the work- place, especially close friends, can influence a person's decision to take a certain job [5, 6]. There is a general consensus among scholars that the peers of young people influence the type of jobs they get. Literature review shows that peer decisions have a dramatic influence on the

decisions that teens make regarding jobs. Friends influence people since they are social beings that do things together and make people feel welcomed and accepted. Another study found out that peer pressure is the second important factor apart from parental and educational factors that influence the professions of young people particularly the female. Much of the time, friends are sought for assistance and guidance especially when they need to make crucial decisions such as the kind of job to do. Teachers, parents and lawmakers should be aware that peers influence decisions made in professional capacities in some ways. By identifying what they do, it may enable stakeholders to facilitate one's support network to assist youths achieve their employment objectives. Building good social relations among peers, stimulating job search within groups, and maintaining good relations among peers are important strategies to assist adolescents to embrace healthy and constructive career decisions [7].

2.3 School Environment and Guidance Counseling

How do classroom dynamics and career counseling influence students' choices? This highlights the critical need for teachers to assist students in making informed choices about their future careers. The importance of school counselors providing students with professional presentations, seminars, and informational events as well as individual career counseling is highlighted in the study. School career counselors can help students find meaningful work by advising them on course management, conducting interest assessments, planning and executing field trips to relevant businesses, presenting career seminars, and teaching courses on academic planning and personal/social development. The study established that school counselors cannot have significant effects on the career choices by students. This could be because they do not have enough resources, do not follow up on the students, or do not get in touch with their parents. All these would prevent students from picking the best majors and occupations. The research discovered that the student intervention programs which are in a habit of assisting these students to make the right decisions were effective in dealing with the problems and improving the ability to decide at work [8]. Career counseling, to some extent done by school counselors, may lead the student to success in one's career and can make better decisions related to future employment. According to the research, college tutors needed greater training and availability of course materials to enable their success with students; the study also pointed out how significant the educational environment was to a student's career choice. It is in the same way that professional advisory programs which are parts of the school provide students with skills and ensure that whatever they are being taught will be used in their future careers. The school environment and guidance counsel play a major role in shaping students' choice of major and eventual job. Schools can help students make better job decisions only if the counselling function is enhanced, good career assistance is offered, and parental influence is taken into account.

2.4 Personal Interest and Skills

A variety of elements, including an individual's hobbies and abilities, have a significant impact on their career pathways, influencing their future selections. Personal interests include whatever a person is enthusiastic about, finds meaning in, or is involved in. Skills, on the other hand, are attributes that people naturally possess and can further develop via life experiences. Prioritizing one's own interests and strengths when choosing a career route is critical. If a person's job choice is in line with their interests and talents, they are more likely to succeed in that position. People are more invested, motivated, and successful when they seek occupations that match their interests, according to researchers. Making informed professional decisions requires a complete assessment of one's talents and abilities. Recognizing one's abilities and areas of expertise allows one to create a professional path that maximizes one's talents while also offering up new chances. It is feasible that people might discover a better fit between their chosen careers and their unique abilities if they matched those abilities to the specific criteria of various occupations [10]. Individual interests and abilities strongly impact professional decisions. If a person's professional choice aligns with their interests, abilities, and personality, they are more likely to succeed in that position. To make well-informed employment decisions that give long-term enjoyment and success, one must first understand oneself well enough to recognize one's passions and abilities. People's hobbies and abilities influence their employment choices significantly since they have a direct impact on work enjoyment, performance, and career success [11].

2.5 Socioeconomic Background

The socioeconomic status of people plays a great role in determining the career they would wish to undertake. Studies show a very close relationship between education and occupation of parents with that of their children. Better-educated parents and those in more influential occupations make use of the resources available to them and provide better mentorship to their children, hence influencing them in choosing a particular career path. Financial disabilities may restrict individuals from accessing certain areas of studies, particularly where a lot of money is required. High social background means that the individual will probably have a bigger social network and connections, which might then avail important information regarding career opportunity. The level of socioeconomic background can influence the perception of career prestige as well as social status, which in turn influences a career choice. Moreover, cultural and social capital, like knowing cultural norms or being connected socially, come into play while making career-related decisions. The inequalities of educational and career opportunities are also predisposed by the differences in socioeconomic background, thus creating disparities in career paths. Recognizing these influences and their potential to guide efforts provides a foundation to create more equitable opportunities and support systems for individuals from diverse socioeconomic backgrounds [12] (Fig. 1).

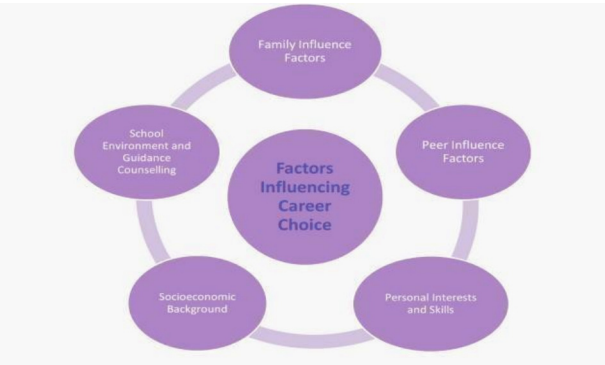


Fig. 1. Factors Influencing Career Choice

3 Gender Differences in Career Aspirations

There have been several studies on the differences in professional aspirations between genders. Studies show that women generally have more ambitious job goals compared to men. In comparison to men, females are more likely to have careers in most industries. Men are more likely to take up careers in male-dominated industries. Professional aspirations can also be influenced by personal preferences, cultural conventions, and gender roles in the workplace. According to some studies, women favor industries with a high concentration of female employees. Consequently, they are not sufficiently represented in the ranks of administration, which is most commonly occupied by men. Alternative careers taken up by women are also encouraged by a better quality of job benefits and opportunities for self and professional development. Establishing gender equality at work and creating an inclusive environment in which every person could pursue his/her professional goals without barriers imposed by stereotypes of who he or she is supposed to be primarily depend on the acknowledgement of differences and overcoming them [13]. The following research evidences the impact of gender-based stereotypes on the career aspirations, selfperception, and confidence of students. Women might choose fields with a higher success likelihood due to stereotypes about gender in certain fields, rather than dominant male successful fields. The scarcity of renowned female role models in crucial professions, especially higher education, might have an impact on the career goals of female students. Role modeling influences and molds pupils' aspirations and potential job opportunities. The study revealed that students' beliefs of the conventional masculinity or femininity associated with certain occupations significantly influence their career aspirations. While male students showed a preference for occupations linked to masculinity, female students opted for fields connected with femininity. Students' work decisions are influenced by gender stereotypes. The findings indicate that in order to motivate students, especially females, to select non-traditional occupations, educational and career counseling settings must address and eliminate gender prejudices. One method to increase the percentage of female students pursuing physical science and related disciplines is by improving technical education in high school [14]. The study identified significant gender disparities in career ambitions. Men exhibited a

preference for careers in STEM and physical sciences, whilst women showed a preference for work that involves interpersonal interaction. Work objectives differ dramatically in countries where women have more political influence. However, this disparity is not universally observed. The gender gap has expanded in countries where women and girls have increased access to political, economic, and educational possibilities. According to the "gender equality paradox," countries with higher levels of female empowerment also had larger disparities in career goals between men and women. Money indirectly contributed to the explanation of the conflict by empowering women and liberating students from the financial limitations imposed by their parents, enabling them to pursue their career goals. The findings indicate that when tackling gender disparity in the workplace, it is crucial to consider both cultural and physiological issues. Gaining insight into the roots of the disparity in career goals between genders might be valuable for formulating effective policies and promoting gender equality in professional decision-making [15].

4 Preferences by Gender of Career Choice Among Secondary School Students

Because there were 101 responses, the following shows how the respondents are spread out by area based on the numbers. The 2% of comments that are about farming were only made by two people. Five people answered "acting," which is 10% of all the answers. Six of the comments, or 5.9%, were about teaching. Two of the people who answered were lawyers, which is 2% of the total. The business sector is the largest part of the company. It is made up of departments like banking, accounting, and marketing. This includes 39 people, or 38.6% of the whole. One in fourteen people, or 14% of the total, came from the Army area, which includes the Navy, Pilot, and Police divisions. Nine people, or one eighth of the total, who answered were medical workers (doctors, nurses, and pharmacists). In the "sports" group, 21 people answered, which is 41.8% of the total. Finally, Politics and Newscasting got 1% of all the answers, with each question getting one answer (Fig. 2).

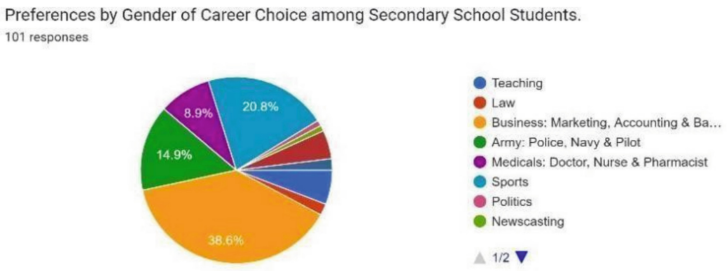


Fig. 2. Preferences by Gender carrier choice among Secondary School students

5 Cultural and Societal Influences on Career Choice References by Gender of Career Choice

A person's career path is significantly influenced by their cultural and social milieu. Decisions regarding one's future and one's perception of potential professional pathways are influenced by numerous factors. The economy, social standards, educational opportunities, and family expectations are all part of this category. The work decision is heavily influenced by family expectations. A person's work performance is frequently influenced by their family's morals and values. For example, certain communities may place a high value on medical professionals and legal professionals, leading individuals to believe they are obligated to carry on the careers of their forefathers. Norms and standards in society also have a role in the case when people choose their careers. Importance that people might place upon social approval, position, and prestige granted from holding a job in a certain sector might lead to interest in that line of career. The major factor pertaining to influencing people's career is the availability of relevant resources and education. People who are less educated or experienced might believe they need to compromise and work in those fields where sufficient opportunities are available. Economic factors, such as the availability of jobs, wages that can be earned, and the level of security connected to a particular job, are some of the factors that contribute to shaping a personal sense of career. During hard times, for example, a worker may be very appreciative of having a job that provides consistently for him or her. Many factors influence people's conceptions of potential occupations, including cultural narratives, social trends, and the media's portrayal of various occupations. Media portrayals of certain occupations can significantly influence people's career aspirations and decisions. Everyone from a person's role models to their coworkers can influence a person's professional trajectory. Anyone interested in learning more about a career path can do so by reading accounts written by those now employed in the area or by consulting those closest to them for advice and following it [16]. When people are making career decisions, cultural and societal considerations are major considerations. Many factors, including cultural standards, familial influences, financial stability, and individual agency, influence women's educational and occupational aspirations. The roles that women are expected to play and the support they get can be shaped by societal and cultural standards. One example of a traditional gender role is the expectation that women should be the primary caregivers and housekeepers. For women, this might mean less opportunities to further their education and climb the corporate ladder. There is a general belief that some jobs are naturally supposed to be performed more competently by men than women. This may be frustrating for women as they may feel less valuable in families whose principal decisions are made by males. The education level and career of a parent also contribute significantly to determining the type of job that their children will be up to. In most researches, it has been established that parents' education level influences the amount of work that their kids would like to do. The level of education from both parents is also another major influence on the careers their female children choose. There is also evidence that shows that the occupational choice made by individuals is primarily under the influence of socio-economic status. The logic behind this argument is those people from the wealthier class have more job opportunities and are better placed in their earlier lives to reach higher levels within their chosen jobs or even careers. Girls' access to

employment or opportunities for further education, as well as schooling, can be more difficult because of cultural and social norms. This is particularly the case in girls' education. The fact that girls in urban areas may still face barriers to achieving their best at school and in later life substantiates the fact that persistent gender stereotypes tend to make housekeeping more important than education, having financial constraints as well as lack of awareness of the options available for jobs. This is crucial for understanding how cultural and socioeconomic factors can be changed to advance gender parity within education. Societies must question past gender norms and allow, if not facilitate, girls to pursue their educational and career aspirations if they are to create a diverse and inclusive workforce that will draw on the skills and contributions of women from every sector. With the personal values, interests, and aspirations that you have in life, it is prudent that you be aware of the cultural and societal forces that you have to navigate around in order to establish the best career. The onus is on the for matter to manufacture these components to exacting standards [17]. The aim of this research is to focus on factors like contextual search because of that user responses and recommendation systems for suggesting ideas. In recent years social media platforms like twitter, Facebook, Instagram etc. [18]. Algorithms with the use of artificial intelligence have ability to convert generic search into satisfying customized searching. Through comparative analysis, it aims to gather data on the efficacy and use of various algorithms and techniques and their impact on the trip search process. For enhancing search in contextual manner, it will be focusing on the specific algorithms [19].

6 Conclusion

Generally, the study brings out wide variation in factors that affect secondary and upper secondary school students' career choice. It actually looks at how the decisions of the people are influenced by the independent and dependent variables. Students should exhibit high self-efficacy so that they can make an appropriate choice on careers. This self-efficacy relates closely to their intentions about the exploration of various careers and the available opportunity in the environment. The current research used both qualitative and quantitative methods of measuring the difference in genders, interests, family situation, educational opportunities, and the world of work standards. The study results generate paramount information regarding designing career counseling and educational interventions which are effective. It is established that students' career goals are critically influenced by their family, peers, and educational setting. The study raised varied issues and barriers to students underpinning their choice of career path: lack of self-awareness, poor research, cultural and familial expectations, emotional instability, and fear of failure. In this regard, a holistic approach involving active learning, engagement of parents and teachers, opportunities for questioning and self-discovery, and availability of relevant resources is a prerequisite for all. In conclusion, by tackling these complex and challenging elements, stakeholders may effectively help kids choose fulfilling occupations.

References

1. Bajwa, J.K., Abai, M., Kidd, S., Couto, S., Akbari-Dibavar, A., McKenzie, K.: Examining the intersection of race, gender, class, and age on post- secondary education and career trajectories of refugees. Refuge **34**(2), 113–123 (2018). https://doi.org/10.7202/1055582ar
2. Pascual, N.T.: Factors affecting high school students' career preference: a basis for career planning program. Int. J. Sci. Basic Appl. Res. (IJSBAR) **16**(1), 1–14 (2014). http://gssrr.org/index.php?journal=JournalOfBasicAndApplied
3. Doshi, M.: A comparison of data mining approaches for forecasting sales of FMCG food products. In: 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1–7. Delhi, India (2023). https://doi.org/10.1109/ICCCNT56998.2023.10307432
4. Swinhoe, K.: Factors affecting career choice among full-time students in a college of commerce. Vocat. Asp. Educ. **19**(43), 139–154 (1967). https://doi.org/10.1080/03057876780000161
5. Hirapara, J., Vanjara, P.: A comparative study of data mining techniques for agriculture crop price prediction. In: 2022 IEEE 7th International conference for Convergence in Technology (I2CT), pp. 1–6 Mumbai, India (2022). https://doi.org/10.1109/I2CT54291.2022.9824533
6. Akosah-Twumasi, P., Emeto, T.I., Lindsay, D., Tsey, K., MalauAduli, B.S.: A systematic review of factors that influence youths career choices—the role of culture. Front. Educ. **3**. Frontiers Media S.A. (2018). https://doi.org/10.3389/feduc.2018.00058
7. Amoah, S.A., Kwofie, I., Akosua, F., Kwofie, A.: J. Educ. Pract. www.iiste.org ISSN (2015). www.iiste.org
8. Amit, K., Patel, A., Atkotiya, K.H.: A review paper on SEO for ranking and effectiveness techniques in context of google search engine, paripex-indian. J. Res. **9**(1) (2015). www.iiste.org
9. Njeri, M.B.: Factors influencing career choices among undergraduate students in public universities in kenya-a case of compassion international sponsored students a research project report submitted in partial fulfilment of the requirement for the award of degree of master of arts in project planning and management of the university of nairobi (2013)
10. Mirzanaveedshahzad, S., Takdees, F., Ahmed, M.: Factors Influencing on Students' Career Choices: Multivariate Analysis (2018). www.ijcar.net
11. Solanki, M.D., Gujarati, P.: The digital revolution in recruitment: unraveling the impact and challenges of e-recruitment. Educ. Adm. Theory Pract. **30** (2024). https://doi.org/10.53555/kuey.v30i6(S).5362
12. Atieno Obura, C., Odongo Ajowi, J.: Academic Journal of Interdisciplinary Studies Gender and Students' Perceptions of Career Aspirations in Secondary schools (2012). https://doi.org/10.5901/ajis.2012.v1n2p149
13. Stoet, G., Geary, D.C.: Sex differences in adolescents' occupational aspirations: Variations across time and place. Plos One **17**(1) (2022). https://doi.org/10.1371/journal.pone.0261438
14. Wicht, A., Miyamoto, A., Lechner, C.M.: Are girls more ambitious than boys? vocational interests partly explain gender differences in occupational aspirations. J. Career Dev. **49**(3), 551–568 (2022). https://doi.org/10.1177/0894845321991665
15. Ann assessment of factors innuencing career choices among fresh graduates: a survey off students majoring businessman economics.
16. Jemini-Gashi, L., Kadriu, E.: Exploring the career decision-making process during the covid-19 pandemic: opportunities and challenges for young people. Sage Open **12**(1) (2022). https://doi.org/10.1177/21582440221078856
17. Bullock-Yowell, E., McConnell, A.E., Schedin, E.A.: Decided and undecided students: career self-efficacy, negative thinking, and decision making difficulties. NACADA J. **34**(1), 22–34 (2014). https://doi.org/10.12930/nacada-13-016

18. Doshi, M., Hirapara, J.: A study on data mining techniques for forecasting FMCG product sales. In: 2024 15th International Conference on Computing Communication and Net- workingTechnologies(ICCCNT), pp.1–6.Kamand,India(2024).https://doi.org/10.1109/ICCCNT61001.2024.10726077
19. Vachhani, P., Parsana, F.: A Comparative study of a hybrid recommender system merging collaborative and content-based strategies for nutritional tailoring in children. 381–386 (2024). https://doi.org/10.1109/IC2SDT62152.2024.10696128
20. Kiritbhai, K.M., Ranpara, R.: A comprehensive comparative analysis of artificial intelligence-based recommender system algorithms for enhancing travel search experience. In: Goar, V., Sharma, A., Shin, J., Mridha, M.F. (eds.) Deep Learning and Visual Artificial Intelligence. ICDLAI 2024. Algorithms for Intelligent Systems. Springer, Singapore (2024). https://doi.org/10.1007/978-981-97-4533-3_17
21. Kiritbhai, K.M., Ranpara, R.: Analyzing the impact of social media dynamics through sentiment analysis: a case study on influencer-generated content and response on recommendation systems amidst the Lakshadweep-Maldives situation. In: 2024 First International Conference on Pioneering Developments in Computer Science & Digital Technologies (IC2SDT), pp. 375–380. Delhi, India (2024). https://doi.org/10.1109/IC2SDT62152.2024.10696841

Harassment Detection in Online Discourse Using Modified Metaheuristic Natural Language Processing

Stanislava Kozakijevic, Luka Jovanovic, Jasmina Perisic, Nebojsa Bacanin(✉), Milos Mravik, Miodrag Zivkovic, and Milos Antonijevic

Singidunum University, Danijelova 32, 11000 Belgrade, Serbia
stanislava.kozakijevic.20231501@fmk.edu.rs, {ljovanovic, jperisic,nbacanin,mmravik,mzivkovic, mantonijevic}@singidunum.ac.rs

Abstract. The internet provides an outstanding platform for individuals to easily communicate, share interests, and connect on a global scale. This connectivity fosters rich interactions and the exchange of ideas across borders. However, the same factors that enable such positive engagement also facilitate the occurrence of harassment, especially given the anonymity that the internet offers. The negative impacts of online harassment can be profound, often mirroring the harmful effects seen in in-person bullying and mobbing, leading to significant declines in mental health. To combat this issue, artificial intelligence (AI) offers a promising solution due to its capacity to quickly process and analyze extensive datasets. While AI is commonly employed with numerical data, natural language processing (NLP) techniques are specifically designed to handle textual data, making them ideal for detecting harassment in online communications. This study explores the potential benefits of integrating optimization metaheuristics with the AdaBoost classifier to improve the detection of abusive comments on YouTube. To enhance the effectiveness of this approach, a modified version of a well-established metaheuristic is introduced. The model optimized with this modified approach demonstrates promising results, achieving an accuracy of 0.904762, indicating its potential for effective harassment detection.

Keywords: Harassment detection · Optimization · Natural language processing · Social media · Abuse detection

1 Introduction

The internet offers a great chance for people to easily communicate, share interests and content, and connect globally. There are many uplifting stories of old friends maintaining friendships across the world, people meeting and becoming close over shared interests, learning about other cultures and building meaningful and tolerant intercultural relationships. Unfortunately, on the flip-side of the ability to interact with so many people,

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 350–363, 2025.
https://doi.org/10.1007/978-3-031-86296-0_25

alongside easy access to anonymity, is that it enables harassment. Harassment comes in many forms online, some of the notable examples being comments and private messages with offensive, hateful, and hurtful contents, purposeful provocation, stalking or sharing someones private information without consent [3].

Being on the receiving end of such negative behaviour can lead to a decline of mental health similar to in-person bullying and mobbing consequences [21]. Additionally, harassment can encourage privacy violations, fear of sharing opinions, financial losses, users leaving platforms and in general a hostile community. It can be hard to correctly identify harassment due to the perpetually changing language and slang [4]. Additionally, considering the vastness of the internet spaces, detecting harassment manually is unfeasible. Artificial intelligence (AI) can make the process easier, with its ability to quickly adapt and analyze large amounts of data. While AI traditionally works with numerical data, a natural language processing (NLP) approach can be implemented to detect harassment. In order to achieve this shift from numerical data to language, the data must be encoded, and one of the available methods is bidirectional encoder representations from transformers (BERT).

When AI is applied to data that has high complexity, it is important that the model choose the most relevant hyperparameters. However, due to the mentioned complexity this is a nondeterministic polynomial time (NP-hard) issue, and manually choosing these hyperparameters is inadequate. A better approach has emerged in the form of metaheuristic optimisers. Many optimisers are available, each with their own strengths and weaknesses. This is highlighted in the no free lunch (NFL) theorem [23], justifying the search for the best suited algorithm to the particular problem.

This work seeks to explore the potential of coupling optimization metaheuristics with the AdaBoost classifier to improve detection of harassing comments on YouTube. A modified version of the genetic algorithm (GA) is proposed that overcomes some of the observed limitations of the original and further boost performance. The introduced detection methodology is evaluated on a real-world dataset and subjected to a comparative analysis against other well established optimizes. The scientific contributions of this work can be summarized as:

- A proposal of a methodology for harassment detection based on NLP techniques and machine learning (ML) classifier.
- An introduction of a modified version of the GA that overcomes some of the observed limitations of the originals.
- A comparative analysis between contemporary optimisers to determine a suitable approach for harassment detection optimization

The reminder of this work obeys the following structure: Sect. 2 discusses preceding research in related topics and highlight the research gap addressed in this work. Section 3 presents the introduced modified algorithm and discusses some of the observed advantages over the basic approach. In Sect. 4 and Sect. 5 the experimental configuration and outcomes are described respectively. Future works and the limitations of this study are finally discussed in Sect. 6 where the work is concluded.

2 Related Works

NLP is a field of AI that enables algorithms to successfully analyze, interpret and respond to humans using natural languages in everyday occasions, despite AI primarily communicating through using numbers. Such AI has a wide array of uses in everyday life as humans have more and more need for computer interactions, some applications including automatic translations, smart phone AI assistants, text generation, customer service.

Due to the aforementioned NFL theorem, there is a need for various algorithms to exist as well as for research to be conducted in order to find the niche for each of them, and the best suited algorithm to each problem. Metaheuristic algorithms are often inspired by animal behaviours, widely known examples being particle swarm optimisation (PSO) [15], bat algorithm (BA) [24], and reptile search algorithms (RSA) [1]. A subsection of algorithms can also be considered the insect-inspired algorithms such as Artificial bee colony (ABC) [14] and Butterfly organisation algorithm. However, some scientist found inspiration in more abstract ideas, and thus we have sinh cosh algorithm (SCHA) [2] inspired by the field of mathematics, genetic algorithm (GA) [18] from the biology field, and variable neighbourhood search (VNS) [9] connecting everyday living conditions to science. An interesting optimizer is also the Botox optimization algorithm (BOA) [11] inspired by cosmetic procedures is another example of a unique inspiration.

Optimisation of hyperparameters using metaheuristic algorithms can make significant difference in the accuracy and efficiency of a model, as it can help guide the model to the most relevant and useful variables. Thus, hyperparameter optimisation could, in the specific case of harassment detection, help the model focus and pay more attention to the relevant words, or even sentence structures that are especially likely to be written with malicious intent. A popular method of improving these algorithms further is by hybridizing them, or joining the best features of multiple algorithms into an advanced, specialised unit.

Examples of metaheuristic algorithms and their hybrid offshoots in science reach many field, thus can be used to help with medical diagnostics [13, 17,22]. They have also found use in space weather forecast [12] and renewable energy generation prediction [6, 19]. Moreover, they have been applied to online security issues [8, 20]. Even employee satisfaction analysis has seen benefit from the method [16].

Despite online harassment being an everyday problem with real-life consequences, the literature available on the topic is limited. As every platform has its own way of functioning and building community and interactions, research targeting specific popular platforms is needed. Harassment in Youtube comments, identified by implementing BERT and AdaBoost methods especially suited for the problem is a topic not yet covered in previous papers, and will thus be the topic of the manuscript.

2.1 Bidirectional Encoder Representations from Transformers

When writing code, mistakes such as spelling and order of operations can cause the code to not only not work, but potentially endanger the rest of the project. The interpret compiler can aid with these smaller errors, for some programming languages. However, with the rise in complexity of code and size of data bases, this process can not only be time-consuming, but also inadequate. Subtler mistakes such as allocating and protecting

memory and those regarding the logic of the code are rarely identified by the traditional approach. Tools that can go beyond mere instruction following, and instead analyze the purpose and idea behind the code could greatly improve the work of programmers.

When processing human language, ML models are at a disadvantage, as they typically work with numerical data. In order for a ML model to be applied to NLP, the text that needs to be analyzed must first be transformed into language that ML models understand. One approach that is gaining attention is BERT [7]. By implementing a transformer as the main ingredient of the encoder, BERT focuses its attention processes to different data portions, effectively splitting its attention on several tasks at once. This allows for higher efficiency, as multiple difficult tasks can be performed at once. In addition, BERT can interpret more complicated ideas since it doesn't focus simply on analysing each word separately. This is achieved through the use of bidirectional data flow. Thus, BERTs ability to interpret subtler emotions and meanings in text has allowed for its wide use in NLP.

BERT models are trained on text in which random words are hidden, and the models job is to make a prediction what the hidden words are. This is a technique taken from masked language models, and it allows for the model to learn the structure and pattern of the language its working with. If its trained on a large dataset, BERT shows transfer learning, using the knowledge from its training and applying it to new situations. Once trained well, the model can be used for classifying text, summarizing it, or answering questions about it, making it a crucial tool for smart systems.

Applied to the problem of harassment through YouTube comments, BERT can transform each comment into a format the ML model can analyse, by utilising it ability to understand the context and deeper meaning behind each comment. This is done through the aforementioned procedure, allowing BERT to break comments into smaller, meaningful units, while retaining semantic information provided. By analyzing the preceding and following text, BERT can detect harassment and such meaningful nuances as sarcasm, informal language, and specific slang used in YouTube comments, alongside the more obvious curses and insults.

2.2 AdaBoost

AdaBoost, as Adaptive boosting is known, is the name of a common ML method in which a classifier is made up of multiple inadequate classifiers [10]. The classifiers used for combining give slightly better results than chance, however not good enough to be standalone algorithms. The first step in the training is to assign the weights for all instances:

$$w_i^{(1)} = \frac{1}{N} \quad \text{for} \quad i = 1, 2, \ldots, N \tag{1}$$

the N symbol marks the number of instances in the training set. The group of weaker classifiers is trained by repetition. For every t iteration, the weaker classifier $h_t\ (x)$ gets training with the current weights $w_i^{(t)}$. Following that is the calculation for the classifier error:

$$_t = \sum_{i=1}^{N} w_i^{(t)} I(y_i \neq h_t(x_i)) \tag{2}$$

here $I(\cdot)$ stands for the indicator function which remains 1 if the condition is true, and 0 if it is false. The calculation of classifier weights is next:

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right) \tag{3}$$

The weights of the instances are adjusted by:

$$w_i^{(t+1)} = w_i^{(t)} \exp(-\alpha_t y_i h_t(x_i)) \tag{4}$$

then normalized to achieve the sum of weights being 1:

$$w_i^{(t+1)} = \frac{w_i^{(t+1)}}{\sum_{j=1}^{N} w_j^{(t+1)}} \tag{5}$$

In the end, the weak classifiers are merged to create a better classifier H(x)$H(x)$ which is the weighted sum of the decisions made by all the weak classifiers:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \tag{6}$$

AdaBoost is especially useful for binary classification, as it can reach great performance by utilizing weak classifiers. Since its focus is on the challenging samples and gives less attention and value to the easy samples, overfitting is not a problem for the model, even when it comes to unbalanced datasets. Its downsides do include a heightened reactivity to noise, need for computational power for the repetitive processes, and prefers balanced data, since the weights need to not be adjusted based on rare classes.

Thus, by utilizing the AdaBoost approach,a good binary classification of YouTube comments can be made, whether it is identifying harassing vs not harassing comments as in this research, or positive or negative comments (sentiment analysis), or harmful vs harmless comments for future works. Since AdaBoost focuses on the ambiguous, harder to classify comments, allowing it to achieve higher performance.

3 Methods

3.1 The Original GA

The principles of natural selection, genetic inheritance, and spontaneous mutation are crucial ideas of evolution. These principles serve as the main inspiration for the GA [18]. Natural selection dictates that the individuals best suited to their specific environment will have better chances of survival and, further, reproduction making their traits more common amongst the species through genetic inheritance. The only way to introduce new traits in a gene pool is mutation, which happens naturally and can both be beneficial and debilitating to the individual.

Similarly to these processes, GA generates populations of chromosomes which carry the genes, or data the algorithm is working with in this case, and allows for the evolutionary process to choose and hone in the best available solutions for a given task. This is done by evaluation and assigned scores to each unit, marking the fitness of each

solution in regards to the current population. Crossover is used to produce offspring by splitting up existing agent chromosomes from the selected population and joining them in new combinations. The second mechanism introduced mutation replacing the existing values of specific chromosomes in agent with completely new values. The basic systems behind these are provided in Fig. 1

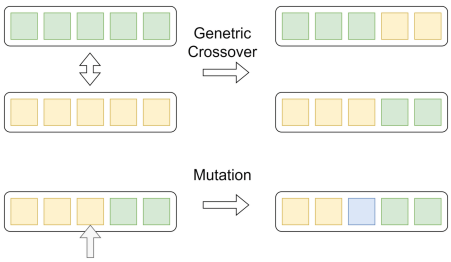


Fig. 1. Crossover and mutation procedures.

These mechanisms are then combine with selection procedures. Solutions are monitored thought the iterations with chromosomes with higher fitness scores having higher reproductive chances, leading to the algorithm favouring more fit solution. Striking a balance between exploration and exploitation is vital for attaining desirable outcomes.

3.2 Proposed Modified GA

The original GA demonstrates decent performance applied to many optimization tasks. However, the potential of this optimizer in combination with recently proposed mechanisms for boosting the performance of mathematics has yet to be explored. This work seeks to boost the diversification potential of the GA by introducing two new mechanisms.

To help strike a more adequate balance between exploration and exploitation the first modification is introduced: the $L1$ norm [5] is added to the original GA. This allows the modified optimizer to adjust for population diversity as well as problem dimensionality. This modification allows for the suitable diversity to be maintained in the YouTube comments harassment dataset, promoting a search for solutions appropriate for such a diverse population, as the comments on social media naturally are. Additionally, the modification helps keep a balance between exploration and exploitation, by on one hand promoting diversity, and on the other focusing on critical features. This way, the model avoids getting stuck on only certain kinds of harassment while ignoring others. Given the population size denoted as m and optimization dimensionality as n the norm can be computed as per Eq. (7–9)

$$\overline{x_j} = \frac{1}{m} \sum_{i=1}^{m} x_{ij} \tag{7}$$

$$D_j^p = \frac{1}{m} \sum_{i=1}^{m\backslash} vlinex_{ij} - \overline{x}_j \backslash vline \tag{8}$$

$$D^p = \frac{1}{n} \sum_{i=1}^{n} D_j^p, \tag{9}$$

here x denotes mean solution positions across all dimensions. Term D_j^p denotes the agents populations diversity as demoted by the $L1$ norm, and D^p represents the populations overall diversity.

A dynamic threshold parameter D_t is introduced to control diversity and reduce convergence in early iterations while boosting convergence in the latter. Worst performing agents are removed form the population as defined by another introduced parameter *nrs*. This parameter defines the number of worst performing agents replaced at the end of an iteration. This term *nrs* is empirically determined to give suitable outcomes when set to $nrs = 2$. Parameter D_t is updated following each iteration to reflect populations diversity.

Should population diversity be determined agents are formatted using uniform crossover and mutation between the poorest solutions and newly generated agents. Otherwise agents are formed as hybrids between the best solution and a randomly selected agent in the population. These mechanisms allow the modified algorithm to take an adaptive approach to an optimisation. As such the opposed algorithm is dubbed the self adaptive GA (SAGA). The procedural code for this method is presented in Alg 1.

Algorithm 1 Procedural code of SAGA

```
Generate population P
Set t = 0
while max_iterations > t do
    Assess each agents in P quality using objective function
    Apply update population using form the GA
    if Diversity not satisfactory then
        Replace the nrs worse solutions with agent combined between a random agent and a newly
        generated solution
    else
        Replace the nrs worse solutions with agent combined between a random agent and a current
        best solution
    end if
    t++
end while
```

4 Simulation Configuration

A comparative analysis is carried out between a select group of contemporary metaheuristics alongside the introduced optimizer. These include the original GA [18] as well as traditionally well established optimizers such as the PSO [15] and VNS [9] algorithms. Swarm optimizers are also included such as the BA [24], ABC [14] algorithms. Some novel optimizers have also been included such as the RSA [1], SCHA [2] and BOA [11] algorithms. Optimizes are independently implemented in Python with parameter settings proposed in the original works that introduced the optimises.

Each optimizer is limited to a population size of six agents with 15 iterations allocated for optimization. Optimizes where tasked with selecting optimal parameters for the AdaBoost classifier form a subset of parameters given in Table 1.

Table 1. AdaBoost hyperparameters and ranges.

AdaBoost parameter range	
count of estimators	[10, 30]
depth	[1, 5]
learning rate	[0.01, 2]

Table 2. Best performing model comparisons in terms of objective function.

Method	Best	Worst	Mean	Median	Std	Var
AB-SAGA	**0.452555**	0.106159	0.291005	0.251273	0.124030	0.015384
AB-GA	0.343066	0.241628	0.296284	0.291908	0.033137	0.001098
AB-PSO	0.371941	0.228275	0.285003	0.257749	0.053009	0.002810
AB-VNS	0.344569	0.257749	0.308974	0.316475	0.033634	0.001131
AB-BA	0.371941	0.120726	0.270981	0.284091	0.089169	0.007951
AB-ABC	0.371941	0.241628	0.287511	0.284091	0.046534	0.002165
AB-SCHA	0.316475	0.257749	0.278102	0.269142	0.022400	0.000502
AB-RSA	0.366589	**0.312909**	**0.336421**	**0.343066**	**0.019744**	0.000390
AB-BOA	0.343066	0.233577	0.298073	0.312909	0.044843	0.002011

Training and evaluations and handled on a real world dataset publicly available on Keggle[1] with this research focusing on the parsed YouTube portion of the dataset. The data is separated in to training and testing portions with the testing portion encompassing 30% while the reminder is used for training. Optimized models are evaluated using standard classifier metrics including accuracy 10, precision 11, specificity 12, f1-score 13.

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \tag{10}$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{11}$$

$$\text{Specificity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{12}$$

$$\text{F1} - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{13}$$

The Cohen's kappa 14 metric is used as the objective for the optimization.

$$\text{Cohen's kappa} = \frac{\text{Model Prediction} - \text{ExpectedClass}}{1 - \text{ExpectedClass}} \tag{14}$$

[1] https://www.kaggle.com/datasets/saurabhshahane/cyberbullying-dataset.

The error rate 15 is tracked as the indicator function.

$$\text{Error rate} = 1 - \text{Accuracy} \tag{15}$$

To aid in understanding the proposed model, a flowchart is included below in Fig. 2

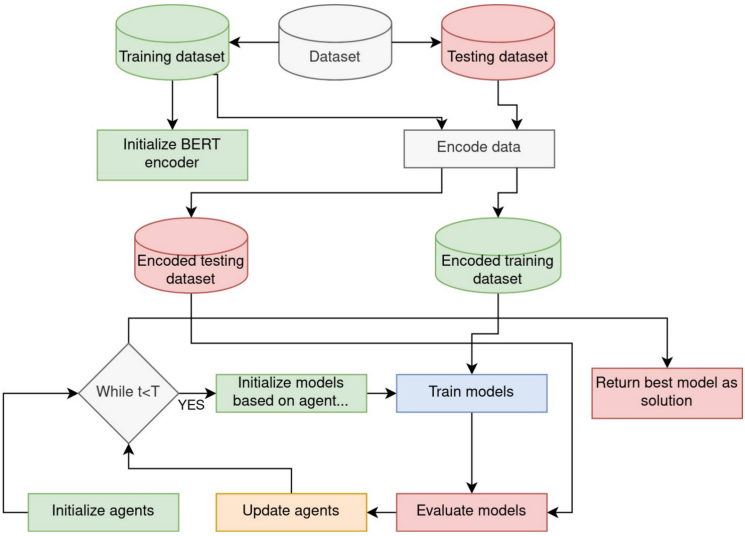


Fig. 2. Model framework

5 Experimental Outcomes

Simulation outcomes in terms of the objective function are provided in Table 2 with the best outcomes highlighted in bold text. The introduced optimizer demonstrated the best performance in the best case execution scenario with a score of 0.452555. However, the RSA demonstrated both admirable stability as well as good outcomes across several execution cases suggesting that this optimizer might be well suited to this optimization challenge.

In terms of indicator function outcomes the introduced optimizer demonstrates the best performance under the best test conditions with a score of 0.095238 and median outcomes scoring 0.104762 as shown in Table 3. The RSA demonstrates the best performance in the mean outcomes as well as a high rate of stability.

Further comparisons between optimizers in terms of stability is provided in Fig. 3. While the introduced algorithm demonstrates the best overall outcomes, in terms of stability a decrease can be observed in comparison to the original. This is often the result of increasing diversification of an algorithm.

Comparisons based on detailed metrics between the best-performing models optimized by each algorithm are provided in Table 4. A definite improvement over the base

Table 3. Best performing model comparisons in terms of indicator function.

Method	Best	Worst	Mean	Median	Std	Var
AB-SAGA	**0.095238**	0.161905	0.131429	**0.104762**	0.0423354	0.001792
AB-GA	0.114286	0.104762	0.118095	0.114286	0.0129187	0.000167
AB-PSO	0.104762	0.161905	0.137143	0.142857	0.0205149	0.000421
AB-VNS	0.095238	0.123810	0.118095	0.123810	0.0129187	0.000167
AB-BA	0.104762	0.228571	0.135238	0.114286	0.0483374	0.002337
AB-ABC	0.104762	0.104762	0.121905	0.114286	0.0184674	0.000341
AB-SCHA	0.123810	0.123810	0.142857	0.123810	0.0295084	0.000871
AB-RSA	0.123810	0.104762	**0.116190**	0.114286	**0.0071270**	**0.000051**
AB-BOA	0.114286	0.133333	0.118095	0.114286	0.0097124	0.000094

optimize can be observed as well as an advantage over other constructed models. However the VNS algorithm suggest a high detection rate for negative comments and a high macro and weighted averages.

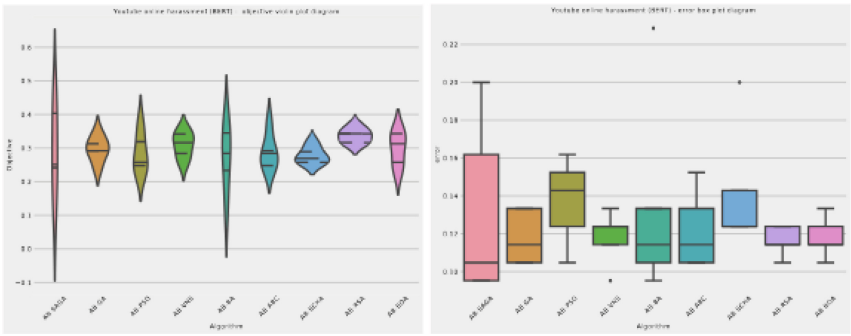


Fig. 3. Objective and indicator function distributions.

Algorithms convergence rates provided in Fig. 4 present a wealth of in formation about an algorithms ability to balance exploration and exploitation. The introduced optimizer avoids local minimum observed to hinder the performance of other optimizers locating the most promising solution in iteration 9 in the case of the objective function. Similarly in terms of indicator function the best outcomes are attained by the introduced optimizer in the ninth iteration while the VNS algorithm locates the same promising score by iteration six. In the case of the VNS this quick convergence comes at a cost of objective function outcomes limiting overall outcome quality.

Finally, parameter selections made by each optimizer for the respective best preforming model are provided in Table 5 so as to better facilitate experimental repeatability.

Table 4. Detailed metric comparison between the best performing models.

Method	Metric	Negative	Positive	Accuracy	Macro avg	Weighted avg
AB-SAGA	precision	0.714286	**0.918367**	0.904762	0.816327	0.893100
	recall	0.384615	0.978261	0.904762	0.681438	0.904762
	f1-score	**0.500000**	**0.947368**	0.904762	0.723684	**0.891980**
AB-GA	precision	0.571429	0.908163	0.885714	0.739796	0.866472
	recall	0.307692	0.967391	0.885714	0.637542	0.885714
	f1-score	0.400000	0.936842	0.885714	0.668421	0.870376
AB-PSO	precision	0.666667	0.909091	0.895238	0.787879	0.879076
	recall	0.307692	0.978261	0.895238	0.642977	0.895238
	f1-score	0.421053	0.942408	0.895238	0.681731	0.877860
AB-VNS	precision	**1.000000**	0.901961	0.904762	**0.950980**	**0.914099**
	recall	0.230769	1.000000	0.904762	0.615385	0.904762
	f1-score	0.375000	0.948454	0.904762	0.661727	0.877455
AB-BA	precision	0.666667	0.909091	0.895238	0.787879	0.879076
	recall	0.307692	0.978261	0.895238	0.642977	0.895238
	f1-score	0.421053	0.942408	0.895238	0.681731	0.877860
AB-ABC	precision	0.666667	0.909091	0.895238	0.787879	0.879076
	recall	0.307692	0.978261	0.895238	0.642977	0.895238
	f1-score	0.421053	0.942408	0.895238	0.681731	0.877860
AB-SCHA	precision	0.500000	0.907216	0.876190	0.703608	0.856799
	recall	0.307692	0.956522	0.876190	0.632107	0.876190
	f1-score	0.380952	0.931217	0.876190	0.656085	0.863089
AB-RSA	precision	0.500000	0.915789	0.876190	0.707895	0.864311
	recall	0.384615	0.945652	0.876190	0.665134	0.876190
	f1-score	0.434783	0.930481	0.876190	0.682632	0.869109
AB-BOA	precision	0.571429	0.908163	0.885714	0.739796	0.866472
	recall	0.307692	0.967391	0.885714	0.637542	0.885714
	f1-score	0.400000	0.936842	0.885714	0.668421	0.870376
	support	13	92			

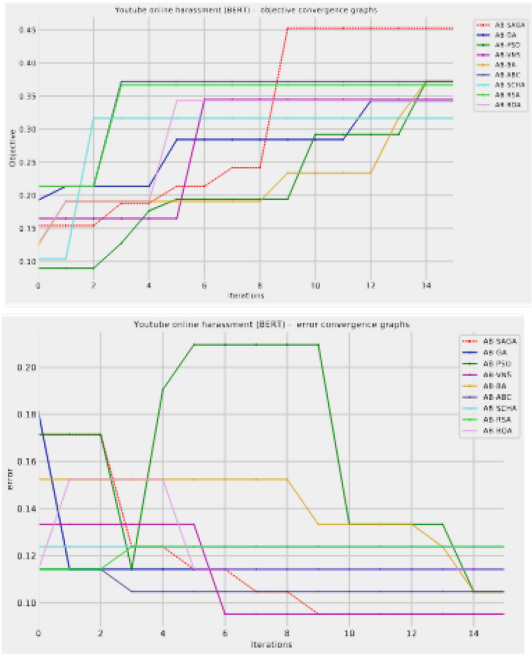


Fig. 4. Convergence diagrams in terms of objective and indicator functions.

Table 5. Best performing model parameter selections.

Method	estimators	depth	Learning rate
AB-SAGA	50	2	2.000000
AB-GA	26	2	0.622649
AB-PSO	32	2	1.615375
AB-VNS	19	1	0.759151
AB-BA	16	1	1.740937
AB-ABC	16	1	1.694690
AB-SCHA	10	3	0.545207
AB-RSA	19	3	0.528523
AB-BOA	50	2	2.000000

6 Conclusion

This work addresses the increasingly pressing issue of harassment detection on online platforms. The rapid shift towards online interactions and the increased adoption of emerging technologies have heightened users' susceptibility to negative online actions. These issues damage communities, make platforms less advertiser-friendly, and negatively impact user mental health. This study explores the potential of optimization metaheuristics coupled with machine learning classifiers for harassment detection through natural language processing. TF-IDF encoding is used to preprocess input data for classifier training and application. Additionally, a modified metaheuristic is introduced to improve per formance through hyperparameter tuning. The modifications introduced allowed for better exploration of the wide range of potential solutions in the hyperparameter tuning space, yielding improved results in comparison to the original algorithm. The outcome of the best-performing model, optimized by the introduced modified optimizer, is promising, with the model demonstrating an accuracy of 0.904762.

The demanding computational requirements of optimization limit the number of optimizers that can be included in the comparative analysis conducted in this work. Additionally, smaller population sizes and limited iterative executions are used due to these constraints. Future work aims to expand this research by including additional algorithms in the comparative analysis. Furthermore, novel applications for the proposed algorithm are expected to be explored in the future.

References

1. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W., Gandomi, A.H.: Reptile search algorithm (rsa): a nature-inspired meta-heuristic optimizer. Expert Syst. Appl. **191**, 116158 (2022).
2. Bai, J., et al.: A sinh cosh optimizer. Knowl.-Based Syst. **282**, 111081 (2023)

3. Blackwell, L., Dimond, J., Schoenebeck, S., Lampe, C.: Classification and its consequences for online harassment: design insights from heartmob. In: Proceedings of the ACM on Human-Computer Interaction **1**(CSCW), 1–19 (2017)
4. Bretschneider, U., Wöhner, T., Peters, R.: Detecting online harassment in social networks (2014)
5. Cheng, S., Shi, Y.: Diversity control in particle swarm optimization. In: 2011 IEEE Symposium on Swarm Intelligence, pp. 1–9. IEEE (2011)
6. Damaševičius, R., et al.: Decomposition aided attention-based recurrent neural net- works for multistep ahead time-series forecasting of renewable power generation. Peer J. Comput. Sci. **10** (2024)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
8. Djuric, M., Jovanovic, L., Zivkovic, M., Bacanin, N., Antonijevic, M., Sarac, M.: The adaboost approach tuned by sns metaheuristics for fraud detection. In: Yadav, R.P., Nanda, S.J., Rana, P.S., Lim, M.H. (eds.) Proceedings of the International Conference on Paradigms of Computing, Communication and Data Sciences. Algorithms for Intelligent Systems, pp. 115–128. Springer, Singapore (2023). https://doi.org/10.1007/978-981-19-8742-7_10
9. Hansen, P., Mladenović, N., Moreno Perez, J.A.: Variable neighbourhood search: methods and applications. Ann. Oper. Res. **175**, 367–407 (2010)
10. Hastie, T., Rosset, S., Zhu, J., Zou, H.: Multi-class adaboost. Statistics and its. Interface **2**(3), 349–360 (2009)
11. Hubálovská, M., Hubálovsky‘, Š., Trojovsky‘, P.: Botox optimization algorithm: a new human-based metaheuristic algorithm for solving optimization problems. Biomimetics **9**(3), 137 (2024)
12. Jovanovic, L., Bacanin, N., Simic, V., Mani, J., Zivkovic, M., Sarac, M.: Optimizing machine learning for space weather forecasting and event classification using modified metaheuristics. Soft Comput. 1–20 (2023)
13. Jovanovic, L., et al.: Tuning extreme learning machine by hybrid planet optimization algorithm for diabetes classification. In: Kumar, S., Sharma, H., Balachandran, K., Kim, J.H., Bansal, J.C. (eds.) Third Congress on Intelligent Systems. CIS 2022. Lecture Notes in Networks and Systems, vol. 613, pp. 23–36. Springer, Singapore (2023). https://doi.org/10.1007/978-981-19-9379-4_3
14. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (abc) algorithm. Appl. Soft Comput. **8**(1), 687–697 (2008)
15. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN’95-International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
16. Kozakijevic, S., Jovanovic, L., Babic, L., Kaljević, J., Zivkovic, M., Bacanin, N.: Machine learning for company review sentiment analysis interpretation. In: Vimal, V., Perikos, I., Mukherjee, A., Piuri, V. (eds.) Multi-Strategy Learning Environment. ICMSLE 2024. Algorithms for Intelligent Systems, pp. 647–659. Springer, Singapore (2024). https://doi.org/10.1007/978-981-97-1488-9_47
17. Kozakijevic, S., et al.: Seizure detection via time series classification using modified metaheuristic optimized recurrent networks. Theor. Appl. Comput. Intell. **1**(1), 82–94 (2023)
18. Mirjalili, S., Mirjalili, S.: Genetic Algorithm. Evolutionary Algorithms and Neural Networks: Theory and Applications, pp. 43–55 (2019)
19. Pavlov-Kagadejev, M., et al.: Optimizing long-short-term memory models via metaheuristics for decomposition aided wind energy generation forecasting. Artif. Intell. Rev.if. Intell. Rev. **57**(3), 45 (2024)
20. Salb, M., et al.: Enhancing internet of things network security using hybrid cnn and xgboost model tuned via modified reptile search algorithm. Appl. Sci. **13**(23), 12687 (2023)

21. Stevens, F., Nurse, J.R., Arief, B.: Cyber stalking, cyber harassment, and adult mental health: a systematic review. Cyberpsychol. Behav. Soc. Netw.tw. **24**(6), 367–376 (2021)
22. Toskovic, A., Kozakijevic, S., Jovanovic, L., Zivkovic, M., Bacanin, N., Antonijevic, M.: Anomaly detection in electroencephalography readings using long short-term memory tuned by modified metaheuristic. In: Pandit, M., Gaur, M.K., Kumar, S. (eds) Artificial Intelligence and Sustainable Computing. ICSISCET 2023. Algorithms for Intelligent Systems, pp. 133–148. Springer, Singapore (2023). https://doi.org/10.1007/978-981-97-0327-2_10
23. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evol. Comput. **1**(1), 67–82 (1997)
24. Yang, X.S., Hossein Gandomi, A.: Bat algorithm: a novel approach for global engineering optimization. Eng. Comput. **29**(5), 464–483 (2012)

Analyzing Extractive Summarization Methods for Kannada Using Sentence Ranking and K-Means Algorithm

Dakshayani Ijeri(✉), Pushpa B. Patil, and Sanjana Mirajkar

Department of Computer Science and Engineering, BLDEA's V. P. Dr. P. G. Halakatti College of Engineering and Technology, (Affiliated to Visvesvaraya Technological University, Belagavi-590018), Vijayapura 586103, Karanataka, India
ijeridakshayani@gmail.com

Abstract. In the world of internet people are dependent on the data available in digital form, due to which millions of information is being generated every day over the internet. The busy schedule of day today's life had made it is a mandatory requirement to have an automation to extract the useful information from the data available over the internet in a concise timeframe. The short description of information can be generated with the help of automatic text summarization. The available systems work efficiently on English language, as India is a diverse country with many languages being spoken across the country and due to which public even use their native languages to convey the information through digital platform. Kannada is one of the prominent languages used by the southern India and 51 million population of all over the world use this language for communication, very less work has been carried out on Kannada language in the domain of Natural Language Processing. The proposed model is implemented on Kannada language for automatic text summarization using extractive method based on sentence ranking algorithm and K means clustering algorithm and got better result for sentence ranking approach with F-score of 0.68 on Rouge 1, F-score of 0.4 on Rouge 2 and F-score of 0.4 on Rouge L over 10,000 documents.

Keywords: K Means Cluster algorithm · Kannada Text Summarization · Sentence Ranking algorithm

1 Introduction

The abundant data is being generated and floated over the digital world every day in all possible languages across the world. Hence it has become significantly necessary to have a model to process these languages in order to understand and interpret for various purposes, for which the domain Natural Language Processing (NLP) has evolved with many amenities. Few of the applications from NLP are sentiment analysis which is used to identify the emotion of the information, chatbot which helps to communicate with clients with Artificial Intelligence (AI) generated answers, machine translation to translate the text from one language to other and text summarization which can be used

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 364–378, 2025.
https://doi.org/10.1007/978-3-031-86296-0_26

to generate the short description from the larger information. All these applications are already implemented on English language with fair performance but very limited work is being carried out on Indian regional languages especially Kannada language. It is quite difficult to model a system to process Kannada language due to the high morphological features of this language, hence very limited NLP work is carried out on Kannada language. Automatic text summarization is helpful to convey the required information in concise explanation in short time which will be major assistance to extract the required content from the larger document. Text summarization is implemented in two ways namely extractive text summarization and abstractive text summarization. Extractive text summarization [1] is a method in which the supreme content from original text is recognized and results as the summary, with contrast to this the abstractive text summarization is a method in which summary is generated by analyzing and interpreting the original text rather than considering the sentences from original text. Limited work is carried on Kannada language for text summarization, but facing with some challenges in performance of summarization. The existing extractive summarization work does not perform well with larger input size and is implemented by considering only few words per sentence, the order of sentences in generated summary may not impart the accurate meaning of the information.

The proposed model is implemented on Kannada language using extractive text summarization method. The system is based on sentence ranking algorithm [3] and K means clustering algorithm. Sentence ranking algorithm uses the Term Frequency (TF) – Inverse Document Frequency (IDF) [2] method to extract features in which total frequency of each word is calculated and upon which importance of word with respect to document is calculated. This TF-IDF feature is used to assign the rank to each sentence and based on which highest rank sentences are selected for summary. K-Means clustering algorithm is experimented with TF-IDF features, in which the clusters are selected using Elbow method and each is sentence is assigned to nearest centroid based on Euclidean distance. Summary is generated by extracting the representative sentence from each centroid. The model had performed better for Sentence Ranking algorithm among two methods.

The further paper is organized with following structure: Sect. 2 covers the related work with respect extractive summarization on Kannada and other Indian regional languages. Section 3 explains about the proposed methodology, Sect. 4 annotates performance measures, in Sect. 5 results and discussions are demonstrated. Section 6 concludes the work with future work.

2 Related Work

In 2023 Anand and Srinivasu [1] had proposed a model for extractive text summarization on Telugu language using graph-based approach. Ranks are assigned based on count of nouns and verbs in a sentence. This system is experimented on 3000 text documents with average F-score of 0.67. The important sentences for summary are based on only nouns and verbs.

In 2022 Pawar et al. [2] proposed a model of extractive text summarization in English language using clustering and word graph approach, the extraction of similar phrases is

computed based Term Frequency and Inverse Document Frequency with the F1-Score of 52%. The accuracy would have been more convincing if discussed with respect to dataset size.

In 2022 Pallavi and Sarvamangala [3] proposed a model for Kannada Text summarization using extractive method based on sentence ranking algorithm. The model is implemented on input with 20 sentences with 256 words and generated output with 90 words. It would have been better if author had discussed with accuracy and dataset.

In 2022 Durga and Babu [4] developed a Telugu text summarization model using Histo Fuzzy C-Means and Median Support Based Grasshopper Optimization algorithm (MSGOA) and achieved accuracy of 84%. Performance of this model is improved with increase in number of iterations.

In 2021 Jayashree et al. [5] had developed a model for Kannada text summarization using POS tagging. This system is based on extractive summarization using sentence ranking algorithm and achieved average accuracy of 0.607. It is experimented on 8 articles with 10 lines each. During POS tagging the system some times categorized POS words as stop words.

In 2021 Jayan and Govindaru [6] proposed a system of text summarization on Malayalam language. This model is implemented using Latent Semantic Analysis (LSA) and Text Rank algorithms and achieved F-score of 66.99 for LSA and 62.32 for Text Rank, these performance measures are based on count of number of unigrams (Rouge 1) matched between system generated summary and reference summary. F-score calculation would have been more justifiable if data size information is available.

In 2020 Jayashree and Vinay [7] developed a model of text summarization on Kannada language using Jaccard's similarity score with 40 documents in different domains. The flow of information in generated summary may have sentences out of chronological order. It would have been helpful if the article consists detailed information regarding data size and accuracy.

In 2020 Castaneda et.al [8] had proposed a model for text summarization on Hindi language using genetic algorithm and achieved Rouge 1 score of 0.1135. It would have been helpful if the article consists detailed information regarding data size.

In 2019 Munzir et al. [9] proposed a model of Text summarization for Bengali language using deep learning and achieved F1 scores of 0.63, 0.59 and 0.56 for Rouge-1, Rouge-2 and Rouge-3 respectively. The performance of this system is good when it is trained with very large training data and more iterations, the result is specific with particular domain.

In 2016 Khan et al. [10] adopted clustered sematic graph approach for multi document abstractive summarization, this model assigns semantic role to analyze semantic structure of text and achieved F-Score 0f 0.61. The system performs better for unigram similarity as compared to bigram similarity.

In 2016 Cheng and Lapata [11] proposed an extractive text summarization using Neural Summarization model. It adopts CNN model for feature extraction. The training data considered in this model contains the prime information in the form of list with bullets which greatly influences the summary generation.

In 2015 Geeta and Deepamala [12] developed a model for Kannada text summarization using latent semantic analysis and got average accuracy of 0.87. This system is

experimented on average 60 sentences of input, for evaluation it considers input with 10 sentences. The performance of LSA decreases with increase in input size and only one meaning of each word is considered.

Following are the Identified Challenges From Existing Work

- The existing work with better accuracy is experimented with limited data size.
- The Sentence rank is calculated based on count of only POS words.
- Flow of information is inappropriate from the order of sentences with generated summary.

3 Proposed Methodology

The process of reducing the original text to short description by extracting the important sentences from input text is known as extractive text summarization. This work proposes the experimentation and comparison of extractive text summarization on Kannada language using Sentence ranking algorithm and K-Means clustering algorithm. Figure 1 demonstrates the methodology of proposed extractive summarization.

3.1 Preprocessing

The Kannada language input text undergoes the initial stage of preprocessing. Following are the steps that helps to preprocess the data:

Tokenize: It breaks the sentence into all possible words known as tokens, for example " ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆ ಯಶಸ್ವಿ" is the input sentence for which tokenizer generates ['ಪರಮಾಣು', 'ಸಾಮರ್ಥ್ಯದ', 'ಖಂಡಾಂತರ', 'ಕ್ಷಿಪಣಿ', 'ಪರೀಕ್ಷೆ','ಯಶಸ್ವಿ']. The English translation of above example is "Nuclear Capable Intercontinental Missile Test Successful" which tokenizer generates ['Nuclear', 'Capable', 'Intercontinental', 'Missile', 'Test', 'Successful'].

Stop Words Removal: The process of excluding the words which actually does not contribute much meaning to the sentence is called stop words removal. Consider the example " ನಾನು ಮತ್ತು ಅವನು ಬೈಸಿಕಲ್ ಓಡಿಸುತ್ತಿದ್ದೇವೆ" for which English translation is "He and I are riding bicycles.", results of stop words removal are " ಬೈಸಿಕಲ್ ಓಡಿಸುತ್ತಿದ್ದೇವೆ" and "riding bicycles" in Kannada and English languages respectively.

Stemmer: This procedure reduces the morphological word to root word. As an example, consider " ನಾನು ಮತ್ತು ಅವನು ಬೈಸಿಕಲ್ ಓಡಿಸುತ್ತಿದ್ದೇವೆ" and in English form "He and I are riding bicycles" for which stemmer generates results as " ನಾನು ಮತ್ತು ಅವನು ಬೈಸಿಕಲ್ ಓಡಿಸು" and "He and I are riding bicycle".

3.2 Word Vectorization

The process of transforming words into numerical representations which can be used to perform the computations in order to process the text through machine learning models is known as word vectorization.

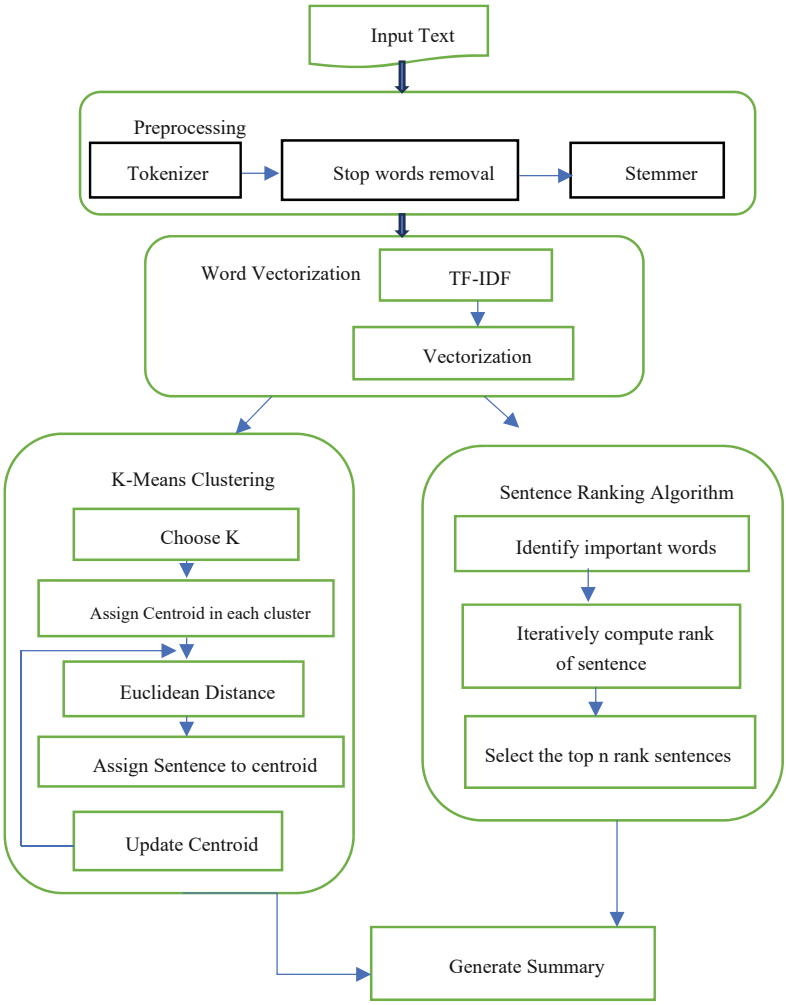


Fig. 1. Methodology of proposed Extractive Text Summarization

Term frequency-Inverse Document Frequency (TF_IDF): This terminology is used to measure the significance of a word with respect to the corpus.

Term Frequency (TF): TF computes the word(w) occurrence rate in a document x as shown in an Eq. (1)

$$TF(w, x) = \frac{f_{(w,x)}}{n(x)} \tag{1}$$

- *f(w,x)* is the word occurrence rate (w) in a document x
- *n(x)* is the complete tally words in document x

Consider the example in Kannada: "*ಕನ್ನಡದಲ್ಲಿ ಟೆಕ್ಸ್ಟ್ ಪ್ರಾಸೆಸಿಂಗ್ ಮಹತ್ವದ ಕ್ಷೇತ್ರವಾಗಿದೆ.*"

Tokenized words: [“"ಕನ್ನಡದಲ್ಲಿ", "ಟೆಕ್ಸ್ಟ್ ", "ಪ್ರಾಸೆಸಿಂಗ್", "ಮಹತ್ವದ", "ಕ್ಷೇತ್ರವಾಗಿದೆ"”].
Consider the calculation of TF of the word “ ಮಹತ್ವದ” from document.
Occurrences of “ ಮಹತ್ವದ”: 1
Tally of words in document:5
TF (“ ಮಹತ್ವದ”, d) = 1/5 = 0.2

Inverse Document Frequency (IDF): IDF computes the significance of a word w with respect to the corpus using Eq. (2)

$$IDF(w) = \log\left(\frac{Total\ number\ of\ documents\ in\ the\ corpus\ C}{1 + |\{x \in C : w \in x\}|}\right) \tag{2}$$

- | {x ∈ C: w ∈ x} |is the total document from corpus where w occurs.
- Adding 1 to the denominator is a smoothing technique to avoid division by zero if the term is not present in any document.

Assuming the corpus with 100 documents and the word “ ಮಹತ್ವದ” appears in 10 documents. The IDF of “ ಮಹತ್ವದ” is:
IDF(“ ಮಹತ್ವದ”) = $\log\frac{100}{10} = 1$
The final TF_IDF is calculated as per Eq. (3)

$$TF_IDF = TF(w, x) * IDF(w) \tag{3}$$

TF of the term “ ಮಹತ್ವದ” in a document is 0.2 (from the earlier TF example) and the IDF is 1 (from the IDF example), then the TF-IDF score is:
TF-IDF (“ ಮಹತ್ವದ”, d) = 0.2*1 = 0.2

Vectorization
Vectorization is the computational representation of the words from the document which is based on the TF_IDF.

$$V(d) = [f_{w1,x}, f_{w2,x}, f_{w3,x}, \ldots\ldots\ldots, f_{wn,x}]$$

- $f_{wi,x}$ is the word (wi) occurrence rate in the document x
- n is the complete tally of words in document x

3.3 Sentence Ranking Algorithm

This algorithm calculates the score of each sentence iteratively from the document using Eq. (4). Top n rank sentences are selected for generating extractive summary.

$$\text{Score}(s) = \sum_{ws} TF_IDF(w, x) \tag{4}$$

S is the sentence in a document x.

Following are the steps for extractive text summarization using sentence ranking method.

```
Input: Document x with list of sentences.
Output: Ranked list of sentences.

Step 1: Compute TF_IDF score of each word w in each
sentence.
  Initialize an empty dictionary for TF_IDF scores
  For each sentence s in document x:
    For each word w in sentence s:
          Calculate Term Frequency (TF) of w in s
          Calculate Inverse Document Frequency (IDF) of
w in document x
          Compute TF-IDF score of w in s
          Store TF-IDF score in TF_IDF_scores[s][w]

Step 2: Compute sentence scores by summing TF-IDF scores
of terms in each sentence
  Initialize an empty dictionary sentence_scores
  For each sentence s in document x:
      Initialize score to 0
      For each term w in sentence s:
            add TF_IDF_scores[s][w] to score
      Store score in sentence_scores[s]

Step 3: Rank sentences based on their scores
  Sort sentences in descending order of their scores

Step 4: Sentence Selection
  Return sorted top n list of sentences
```

3.4 K Means Clustering Algorithm

This algorithm is used to generate extractive text summary. The algorithm initially selects random centroids and assigns each sentence to one of the centroids based on nearest distance vector to form the cluster. This process is carried out iteratively while recalculating centroids for each iteration until all clusters become stable. Finally, each cluster centroid is extracted for summary.

Choose K: Elbow method is used to identify the optimal number of clusters. It executes K means on different k values to calculate sum of squared distances for each k and this is plotted against the number of clusters. Optimal number of clusters is identified from the point where the rate of sum of squared distances decrease sharply slows down. The elbow method applied for proposed work is shown in Fig. 2. Which clearly depicts that optimal number of clusters identified here is 8. Randomly choose initial centroids for identified number of clusters such $c_{1,} c_{2,} \ldots\ldots c_k$ where k is the total number of clusters.

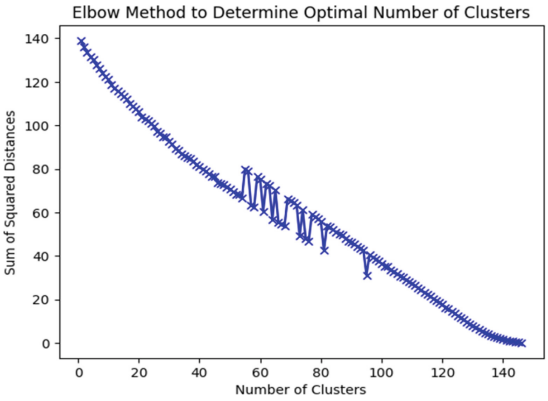


Fig. 2. Elbow Method for optimal number of clusters.

Euclidean Distance and Assignment: Allocate each sentence to the most adjacent centroid depending on the minimum Euclidean distance calculation as per Eq. (5).

$$cl_j = argmin_j \sum\nolimits_i (s_i - c_j)^2 \tag{5}$$

- cl_j is the cluster assigned to the sentence s
- s_i is the sentence vector
- c_j is the centroid of the jth cluster.

Recompute the centroids as per Eq. (6) after the sentences are assigned to the clusters and repeat this process until the centroids remain constant.

$$c_{i(1 \le i \le k)} = \frac{1}{|cl_i|} \sum\nolimits_{s \in cl_i} s \tag{6}$$

- $|cl_i|$ is the entire sentence tally in the cluster i.
- Summation is carried out over all the sentence vectors s_i allocated to the cluster i.

3.5 Summarization

Once the clusters are stable, the sentences that are nearest to the centroids are selected from each cluster for generating summary as per Eq. (7).

$$Summary = \bigcup\nolimits_{j=1}^{k} \{argmin_{s_i \in cl_j} ||s_i - c_j||\} \tag{7}$$

- $\{argmin_{si} ||s_i - c_j||\}$ is the nearest sentence to each centroid from respective cluster.
- $\bigcup_{j=1}^{k} \{argmin_{s_i \in cl_j} ||s_i - c_j||\}$ is the union of selected sentences from each cluster.

Following are the steps for extractive text summarization using K-Means clustering algorithm.

```
Input:
   Text document
Output:
   Summary of the text

1. Preprocessing
   a. Tokenize the text.
   b. Perform TF_IDF on text to express text into numeric
vector representation

2. Select the number of clusters based on Elbow method

3. Centroid Initialization
   a.  Randomly select centroids for each cluster.

4. Iterate until centroids are stable:
   a. Assignment Step:
        - For each sentence vector Si:
        - Calculate the nearest centroid for each
sentence and assign to the cluster:
          cl_j = argmin_j ∑_i (s_i − c_j)^2
   b. Update Step:
      - For each cluster j:
      - Recompute the centroid ci as the mean of all
sentence vectors in cluster j:
          centroids[i] = mean(cluster[j])

5. Validate Cluster
    a. For each cluster j
        - For each sentence vector Si
            -Compute silhouette score
        -Return Mean silhouette score
        - silhouette score range nearer to +1 indicates
better cluster as compared to negative value.

6. Generate summary
   a. For each cluster j:
      - Select the sentence si in cluster j closest to
the centroid ci:
        representative_sentence =
argmin_si_in_cluster_j(distance(si, centroids[i]))
      - Add the representative sentence to the summary:
        summary.append(representative_sentence)
```

7. **Return the summary as a concatenation of selected sentences**

The platform used for implementation is python language. Various packages from python are used at different stages. NLTK package is used for all preprocessing steps, Scikit-learn package for K-means clustering and scipy.spatial.distance for cosine similarity calculation.

4 Results and Discussions

The proposed model is experimented with two methods namely sentence ranking algorithm and K-Means clustering algorithm and results of these methods are discussed in this section.

4.1 Dataset

As of now the standard dataset for text summarization on Kannada language is not available, we have created our own dataset by considering the data from popular Kannada news website https://www.etvbharat.com/. The dataset is constructed with 10,000 documents covering different kinds of domain such as politics, sports, crime etc. appearing in news website. Random sampling is used to select a data from data set allocating equal chance to select each document. The Fig. 3 shows the sample dataset.

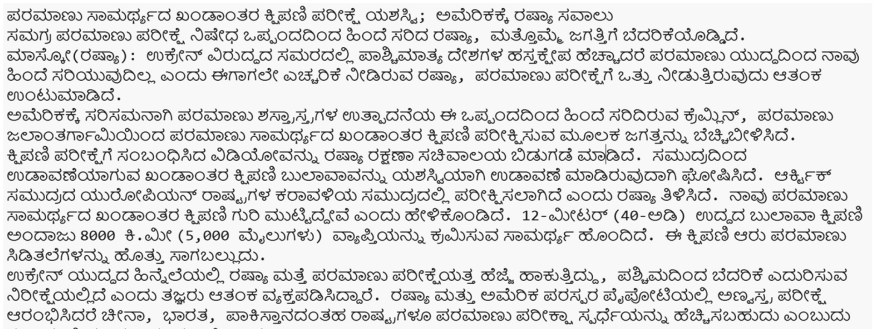

ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆ ಯಶಸ್ವಿ; ಅಮೆರಿಕಕ್ಕೆ ರಷ್ಯಾ ಸವಾಲು
ಸಮಗ್ರ ಪರಮಾಣು ಪರೀಕ್ಷೆ ನಿಷೇಧ ಒಪ್ಪಂದದಿಂದ ಹಿಂದೆ ಸರಿದ ರಷ್ಯಾ, ಮತ್ತೊಮ್ಮೆ ಜಗತ್ತಿಗೆ ಬೆದರಿಕೆಯೊಡ್ಡಿದೆ.
ಮಾಸ್ಕೋ(ರಷ್ಯಾ): ಉಕ್ರೇನ್ ವಿರುದ್ಧದ ಸಮರದಲ್ಲಿ ಪಾಶ್ಚಿಮಾತ್ಯ ದೇಶಗಳ ಹಸ್ತಕ್ಷೇಪ ಹೆಚ್ಚಾದರೆ ಪರಮಾಣು ಯುದ್ಧದಿಂದ ನಾವು ಹಿಂದೆ ಸರಿಯುವುದಿಲ್ಲ ಎಂದು ಈಗಾಗಲೇ ಎಚ್ಚರಿಕೆ ನೀಡಿರುವ ರಷ್ಯಾ, ಪರಮಾಣು ಪರೀಕ್ಷೆಗೆ ಒತ್ತು ನೀಡುತ್ತಿರುವುದು ಆತಂಕ ಉಂಟುಮಾಡಿದೆ.
ಅಮೆರಿಕಕ್ಕೆ ಸರಿಸಮನಾಗಿ ಪರಮಾಣು ಶಸ್ತ್ರಾಸ್ತ್ರಗಳ ಉತ್ಪಾದನೆಯ ಈ ಒಪ್ಪಂದದಿಂದ ಹಿಂದೆ ಸರಿದಿರುವ ಕ್ರೆಮ್ಲಿನ್, ಪರಮಾಣು ಜಲಾಂತರ್ಗಾಮಿಯಿಂದ ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷಿಸುವ ಮೂಲಕ ಜಗತ್ತನ್ನು ಬೆಚ್ಚಿಬೀಳಿಸಿದೆ.
ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆಗೆ ಸಂಬಂಧಿಸಿದ ವಿಡಿಯೋವನ್ನು ರಷ್ಯಾ ರಕ್ಷಣಾ ಸಚಿವಾಲಯ ಬಿಡುಗಡೆ ಮಾಡಿದೆ. ಸಮುದ್ರದಿಂದ ಉಡಾವಣೆಯಾಗುವ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಬುಲಾವಾವನ್ನು ಯಶಸ್ವಿಯಾಗಿ ಉಡಾವಣೆ ಮಾಡಿರುವುದಾಗಿ ಘೋಷಿಸಿದೆ. ಆರ್ಕ್ಟಿಕ್ ಸಮುದ್ರದ ಯುರೋಪಿಯನ್ ರಾಷ್ಟ್ರಗಳ ಕರಾವಳಿಯ ಸಮುದ್ರದಲ್ಲಿ ಪರೀಕ್ಷಿಸಲಾಗಿದೆ ಎಂದು ರಷ್ಯಾ ತಿಳಿಸಿದೆ. ನಾವು ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಗುರಿ ಮುಟ್ಟಿದ್ದೇವೆ ಎಂದು ಹೇಳಿಕೊಂಡಿದೆ. 12-ಮೀಟರ್ (40-ಅಡಿ) ಉದ್ದದ ಬುಲಾವಾ ಕ್ಷಿಪಣಿ ಅಂದಾಜು 8000 ಕಿ.ಮೀ (5,000 ಮೈಲುಗಳು) ವ್ಯಾಪ್ತಿಯನ್ನು ಕ್ರಮಿಸುವ ಸಾಮರ್ಥ್ಯ ಹೊಂದಿದೆ. ಈ ಕ್ಷಿಪಣಿ ಆರು ಪರಮಾಣು ಸಿಡಿತಲೆಗಳನ್ನು ಹೊತ್ತು ಸಾಗಬಲ್ಲುದು.
ಉಕ್ರೇನ್ ಯುದ್ಧದ ಹಿನ್ನೆಲೆಯಲ್ಲಿ ರಷ್ಯಾ ಮತ್ತೆ ಪರಮಾಣು ಪರೀಕ್ಷೆಯತ್ತ ಹೆಜ್ಜೆ ಹಾಕುತ್ತಿದ್ದು, ಪಶ್ಚಿಮದಿಂದ ಬೆದರಿಕೆ ಎದುರಿಸುವ ನಿರೀಕ್ಷೆಯಲ್ಲಿದೆ ಎಂದು ತಜ್ಞರು ಆತಂಕ ವ್ಯಕ್ತಪಡಿಸಿದ್ದಾರೆ. ರಷ್ಯಾ ಮತ್ತು ಅಮೆರಿಕ ಪರಸ್ಪರ ಪೈಪೋಟಿಯಲ್ಲಿ ಅಣ್ವಸ್ತ್ರ ಪರೀಕ್ಷೆ ಆರಂಭಿಸಿದರೆ ಚೀನಾ, ಭಾರತ, ಪಾಕಿಸ್ತಾನದಂತಹ ರಾಷ್ಟ್ರಗಳೂ ಪರಮಾಣು ಪರೀಕ್ಷಾ ಸ್ಪರ್ಧೆಯನ್ನು ಹೆಚ್ಚಿಸಬಹುದು ಎಂಬುದು ಶಸ್ತ್ರಾಸ್ತ್ರ ನಿಯಂತ್ರಣ ತಜ್ಞರ ಅಭಿಪ್ರಾಯ.
ಪುಸ್ತುತ ಈ ಎಲ್ಲಾ ದೇಶಗಳು ಸಹ ಸ್ವಯಂ ಹೇರಿದ ನಿಷೇಧ ನೀತಿ ಪಾಲಿಸುತ್ತಿವೆ. ಪರಮಾಣು ಶಸ್ತ್ರಾಸ್ತ್ರ ಪರೀಕ್ಷೆಗಳನ್ನು ನಿಷೇಧಿಸುವ ಅಂತರರಾಷ್ಟ್ರೀಯ ಒಪ್ಪಂದವನ್ನು ಅನುಮೋದಿಸುವ 2000ರಲ್ಲಿ ಅಂಗೀಕರಿಸಿದ ನಿರ್ಣಯವನ್ನು ತನ್ನ ಸಂಸತ್ತು ರದ್ದುಗೊಳಿಸುವ ಸಾಧ್ಯತೆಯಿದೆ ಎಂದು ಪುಟಿನ್ ಇತ್ತೀಚೆಗೆ ಬಹಿರಂಗಪಡಿಸಿದ್ದರು. ಪರಮಾಣು ಪರೀಕ್ಷೆ ನಿಷೇಧ ಒಪ್ಪಂದಕ್ಕೆ ಅಮೆರಿಕ ಸಹಿ ಹಾಕಿದ್ದರೂ ಆ ದೇಶದ ಕಾಂಗ್ರೆಸ್ ಅದನ್ನು ಇನ್ನೂ ಅನುಮೋದಿಸಿಲ್ಲ ಎಂದು ಅವರು ನೆನಪಿಸಿದರು. ತಾವೂ ಕೂಡ ಅದೇ ನಿರ್ಧಾರ ಕೈಗೊಂಡಿದ್ದೇವೆ ಎಂಬುದು ಪುಟಿನ್ ಮಾತು.
ಉಕ್ರೇನ್‌ನಲ್ಲಿನ ಯುದ್ಧ ಮತ್ತು ಪಶ್ಚಿಮದೊಂದಿಗಿನ ತೀವ್ರ ಉದ್ವಿಗ್ನತೆಯ ನಡುವೆ ರಷ್ಯಾ ಇತ್ತೀಚೆಗೆ ಸಂಚಲನ ನಿರ್ಧಾರ ತೆಗೆದುಕೊಂಡಿತು. ಪರಮಾಣು ಪರೀಕ್ಷೆಗಳ ಬಗ್ಗೆ ರಷ್ಯಾದ ಕಾನೂನು ನಿಲುವನ್ನು ಬದಲಾಯಿಸುವ ಮಸೂದೆಯನ್ನು ಸಂಸತ್ತು ಅನುಮೋದಿಸಿದೆ. ಪುಟಿನ್ ಈ ಹಿಂದೆ ತಮ್ಮ ಅಭಿಪ್ರಾಯಗಳನ್ನು ವ್ಯಕ್ತಪಡಿಸಿದಂತೆ ಪರಮಾಣು ಪರೀಕ್ಷಾ ನಿಷೇಧ ಒಪ್ಪಂದ ರದ್ದುಗೊಳಿಸಲು ರಷ್ಯಾದ ಸಂಸತ್ತು ಮಸೂದೆ ಅಂಗೀಕರಿಸಿದೆ. ಅಮೆರಿಕ 1996ರಲ್ಲಿ ಒಪ್ಪಂದಕ್ಕೆ ಸಹಿ ಹಾಕಿದ್ದರೂ, ಅದನ್ನು ಅಂಗೀಕರಿಸಿಲ್ಲ. ರಷ್ಯಾ ಕೂಡ ಈ ಒಪ್ಪಂದವನ್ನು ರದ್ದುಗೊಳಿಸಿದೆ. ಜಗತ್ತು ಅಮೆರಿಕದಿಂದ ಬೆದರಿಕೆ ಎದುರಿಸುತ್ತಿರುವ ಕಾರಣ ಈ ನಿರ್ಧಾರ ತೆಗೆದುಕೊಳ್ಳಲಾಗಿದೆ ಎಂದು ಪುಟಿನ್ ಹೇಳಿದ್ದಾರೆ.
ಬಿಜೆಪಿಯವರು ಅಂಬಾನಿ - ಅದಾನಿ ಅಂತವರಿಗೆ ಅನುಕೂಲ ಮಾಡುತ್ತಾರೆ: ಸಚಿವ ಕೃಷ್ಣ ಬೈರೇಗೌಡ
ನಮ್ಮ ಸರ್ಕಾರವನ್ನು ಕಿತ್ತೊಗೆಯಲು ಬಿಜೆಪಿಯವರು ಪ್ರಯತ್ನ ಮಾಡುತ್ತಿರಬಹುದು. ಅದನ್ನು ಎದುರಿಸುವ ಶಕ್ತಿ ನಮಗೆ ಮತ್ತು ನಮ್ಮ ಪಕ್ಷಕ್ಕೆ ಇದೆ ಎಂದು ಸಚಿವ ಕೃಷ್ಣ ಬೈರೇಗೌಡ ಹೇಳಿದರು.

Fig. 3. Kannada Input Dataset

4.2 Performance Measures

The proposed model is evaluated with ROUGE (Recall-Oriented Understudy for Gisting Evaluation) which is the set of metrics used for assessing automatic text summarization. The model is measured with three metrics namely ROUGE-1, ROUGE-2 and ROUGE-L. The language experts have helped to generate summaries of the input dataset considered for this model. These summaries written by language experts are the references summaries against with which the model generated summaries are evaluated.

ROUGE-1: Rouge-1 is the value representing the common unigrams between system generated summary and reference summary as per Eq. (8).

$$ROUGE1 = \frac{Count(unigrams\ matched\ between\ Ref_{sum}\ and\ Sys_{sum})}{Count\ (Words\ in\ Ref_{sum})} \tag{8}$$

- Ref_{sum} : *Reference Summary*
- Sys_{sum} : *System generated summary*

ROUGE-2: Rouge-2 is the value representing the common bigrams between system generated summary and reference summary as per Eq. (9).

$$ROUGE2 = \frac{Count(bigrams\ matched\ between\ Ref_{sum}\ and\ Sys_{sum})}{Count(\text{bigrams}\ in\ Ref_{sum})} \tag{9}$$

ROUGE-L: Rouge-L is the value representing the common longest sequence of words between system generated summary and reference summary.

Recall: *Recall for Rouge-L is the ratio of the length of the longest common sequence (LCS) to the length of the reference summary as per Eq. (10).*

$$Recall = \frac{Length(LCS)}{Count(\text{words}\ inRef_{sum})} \tag{10}$$

Precision: *Precision for Rouge-L is the ratio of the length of the longest common sequence (LCS) to the length of the system generated summary as per equation Eq. (11).*

$$Precision = \frac{Length(LCS)}{Count(\text{words}\ inSys_{sum})} \tag{11}$$

F-score: *It is the harmonic mean of recall and precision used to assess the accuracy of comparison between reference summary and system generated summary (12).*

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{12}$$

The Table 1 shows the average Rouge scores of summaries generated over the sample 3 data sets among overall dataset using the Sentence Rank Algorithm (SRA) and K-Means clustering algorithm.

Figure 4 shows the F1-scores of ROUGE-1, ROUGE-2 and ROUGE-L of dataset using Sentence rank algorithm and K-Means clustering method. F1 score of Rouge-1

Table 1. ROUGE scores for Kannada summary using Sentence rank algorithm and K-Means algorithm

	Recall		Precision		F-Score	
	SRA	K-Means	SRA	K-Means	SRA	K-Means
ROUGE-1	0.706	0.814	0.457	0.166	**0.537**	0.283
ROUGE-2	0.296	0.386	0.210	0.075	**0.239**	0.125
ROUGE-L	0.383	0.564	0.248	0.116	**0.292**	0.194

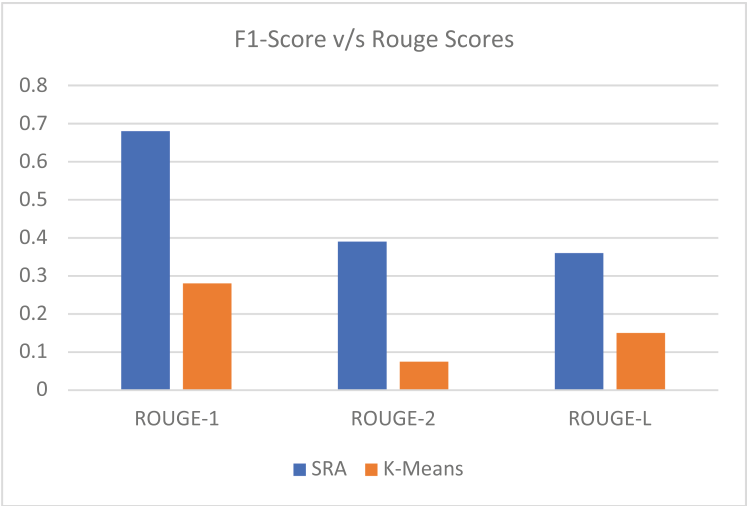


Fig. 4. F1-score values of ROUGE 1, ROUGE 2 and ROUGE L for Kannada Sentence rank algorithm and K-Means Algorithm

is 0.68, Rouge-2 is 0.39 and Rouge-L is 0.36 for SRA. F1 score of Rouge-1 is 0.28, Rouge-2 is 0.075 and Rouge-L is 0.15 for K-Means.

The Fig. 5 shows the input for text summarization in Kannada language. The English translation of first few lines of above dataset is shown in Fig. 6.

Figure 7 shows the summary generated of above input (Fig. 5) using Sentence rank algorithm. The English translation of first few lines of above result is shown in Fig. 8.

Figure 9 shows the summary of above input (Fig. 6) using K-Means cluster method. The English translation of first few lines of this output is shown in Fig. 10.

The proposed model is implemented using Sentence rank algorithm and K-Means cluster method, out of which TF-IDF had performed better with F-Score of 0.292 for Rouge-L. The existing work were developed with a restriction of limited number of lines for input data and accuracy discussed is with very limited dataset, the earlier work on Sentence rank was based on count of parts of speech words and with inappropriate order of sentences in summary. The proposed model had performed better than the existing system. The proposed work does not have restriction on input size and results discussed above are assessed with larger dataset as compared to existing work. The generated summaries convey the meaningful information.

ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆ ಯಶಸ್ವಿ; ಅಮೆರಿಕಕ್ಕೆ ರಷ್ಯಾ ಸವಾಲು
ಸಮಗ್ರ ಪರಮಾಣು ಪರೀಕ್ಷೆ ನಿಷೇಧ ಒಪ್ಪಂದದಿಂದ ಹಿಂದೆ ಸರಿದ ರಷ್ಯಾ, ಮತ್ತೊಮ್ಮೆ ಜಗತ್ತಿಗೆ ಬೆದರಿಕೆಯೊಡ್ಡಿದೆ.
ಮಾಸ್ಕೋ(ರಷ್ಯಾ): ಉಕ್ರೇನ್ ವಿರುದ್ಧದ ಸಮರದಲ್ಲಿ ಪಾಶ್ಚಿಮಾತ್ಯ ದೇಶಗಳ ಹಸ್ತಕ್ಷೇಪ ಹೆಚ್ಚಾದರೆ ಪರಮಾಣು ಯುದ್ಧದಿಂದ ನಾವು ಹಿಂದೆ ಸರಿಯುವುದಿಲ್ಲ ಎಂದು ಈಗಾಗಲೇ ಎಚ್ಚರಿಕೆ ನೀಡಿರುವ ರಷ್ಯಾ, ಪರಮಾಣು ಪರೀಕ್ಷೆಗೆ ಒತ್ತು ನೀಡುತ್ತಿರುವುದು ಆತಂಕ ಉಂಟುಮಾಡಿದೆ.
ಅಮೆರಿಕಕ್ಕೆ ಸರಿಸಮನಾಗಿ ಪರಮಾಣು ಶಸ್ತ್ರಾಸ್ತ್ರಗಳ ಉತ್ಪಾದನೆಯ ಈ ಒಪ್ಪಂದದಿಂದ ಹಿಂದೆ ಸರಿದಿರುವ ಕ್ರೆಮ್ಲಿನ್, ಪರಮಾಣು ಜಲಾಂತರ್ಗಾಮಿಯಿಂದ ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷಿಸುವ ಮೂಲಕ ಜಗತ್ತನ್ನು ಬೆಚ್ಚಿಬೀಳಿಸಿದೆ.
ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆಗೆ ಸಂಬಂಧಿಸಿದ ವಿಡಿಯೋವನ್ನು ರಷ್ಯಾ ರಕ್ಷಣಾ ಸಚಿವಾಲಯ ಬಿಡುಗಡೆ ಮಾಡಿದೆ. ಸಮುದ್ರದಿಂದ ಉಡಾವಣೆಯಾಗುವ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಬುಲಾವಾವನ್ನು ಯಶಸ್ವಿಯಾಗಿ ಉಡಾವಣೆ ಮಾಡಿರುವುದಾಗಿ ಘೋಷಿಸಿದೆ. ಆರ್ಕ್ಟಿಕ್ ಸಮುದ್ರದ ಯುರೋಪಿಯನ್ ರಾಷ್ಟ್ರಗಳ ಕರಾವಳಿಯ ಸಮುದ್ರದಲ್ಲಿ ಪರೀಕ್ಷಿಸಲಾಗಿದೆ ಎಂದು ರಷ್ಯಾ ತಿಳಿಸಿದೆ. ನಾವು ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಗುರಿ ಮುಟ್ಟಿದ್ದೇವೆ ಎಂದು ಹೇಳಿಕೊಂಡಿದೆ. 12-ಮೀಟರ್ (40-ಅಡಿ) ಉದ್ದದ ಬುಲಾವಾ ಕ್ಷಿಪಣಿ ಅಂದಾಜು 8000 ಕಿ.ಮೀ (5,000 ಮೈಲುಗಳು) ವ್ಯಾಪ್ತಿಯನ್ನು ಕ್ರಮಿಸುವ ಸಾಮರ್ಥ್ಯ ಹೊಂದಿದೆ. ಈ ಕ್ಷಿಪಣಿ ಆರು ಪರಮಾಣು ಸಿಡಿತಲೆಗಳನ್ನು ಹೊತ್ತು ಸಾಗಬಲ್ಲುದು.
ಉಕ್ರೇನ್ ಯುದ್ಧದ ಹಿನ್ನೆಲೆಯಲ್ಲಿ ರಷ್ಯಾ ಮತ್ತೆ ಪರಮಾಣು ಪರೀಕ್ಷೆಯತ್ತ ಹೆಜ್ಜೆ ಹಾಕುತ್ತಿದ್ದು, ಪಶ್ಚಿಮದಿಂದ ಬೆದರಿಕೆ ಎದುರಿಸುವ ನಿರೀಕ್ಷೆಯಲ್ಲಿದೆ ಎಂದು ತಜ್ಞರು ಆತಂಕ ವ್ಯಕ್ತಪಡಿಸಿದ್ದಾರೆ. ರಷ್ಯಾ ಮತ್ತು ಅಮೆರಿಕ ಪರಸ್ಪರ ಪೈಪೋಟಿಯಲ್ಲಿ ಅಣ್ವಸ್ತ್ರ ಪರೀಕ್ಷೆ ಆರಂಭಿಸಿದರೆ ಚೀನಾ, ಭಾರತ, ಪಾಕಿಸ್ತಾನದಂತಹ ರಾಷ್ಟ್ರಗಳೂ ಪರಮಾಣು ಪರೀಕ್ಷಾ ಸ್ಪರ್ಧೆಯನ್ನು ಹೆಚ್ಚಿಸಬಹುದು ಎಂಬುದು ಶಸ್ತ್ರಾಸ್ತ್ರ ನಿಯಂತ್ರಣ ತಜ್ಞರ ಅಭಿಪ್ರಾಯ.
ಪ್ರಸ್ತುತ ಈ ಎಲ್ಲಾ ದೇಶಗಳು ಸಹ ಸ್ವಯಂ ಹೇರಿದ ನಿಷೇಧ ನೀತಿ ಪಾಲಿಸುತ್ತಿವೆ. ಪರಮಾಣು ಶಸ್ತ್ರಾಸ್ತ್ರ ಪರೀಕ್ಷೆಗಳನ್ನು ನಿಷೇಧಿಸುವ ಅಂತರರಾಷ್ಟ್ರೀಯ ಒಪ್ಪಂದವನ್ನು ಅನುಮೋದಿಸುವ 2000ರಲ್ಲಿ ಅಂಗೀಕರಿಸಿದ ನಿರ್ಣಯವನ್ನು ತನ್ನ ಸಂಸತ್ತು ರದ್ದುಗೊಳಿಸುವ ಸಾಧ್ಯತೆಯಿದೆ ಎಂದು ಪುಟಿನ್ ಇತ್ತೀಚೆಗೆ ಬಹಿರಂಗಪಡಿಸಿದ್ದರು. ಪರಮಾಣು ಪರೀಕ್ಷೆ ನಿಷೇಧ ಒಪ್ಪಂದಕ್ಕೆ ಅಮೆರಿಕ ಸಹಿ ಹಾಕಿದ್ದರೂ ಆ ದೇಶದ ಕಾಂಗ್ರೆಸ್ ಅದನ್ನು ಇನ್ನೂ ಅನುಮೋದಿಸಿಲ್ಲ ಎಂದು ಅವರು ನೆನಪಿಸಿದರು. ತಾವೂ ಕೂಡ ಅದೇ ನಿರ್ಧಾರ ಕೈಗೊಂಡಿದ್ದೇವೆ ಎಂಬುದು ಪುಟಿನ್ ಮಾತು.
ಉಕ್ರೇನ್‌ನಲ್ಲಿನ ಯುದ್ಧ ಮತ್ತು ಪಶ್ಚಿಮದೊಂದಿಗಿನ ತೀವ್ರ ಉದ್ವಿಗ್ನತೆಯ ನಡುವೆ ರಷ್ಯಾ ಇತ್ತೀಚೆಗೆ ಸಂಚಲನ ನಿರ್ಧಾರ ತೆಗೆದುಕೊಂಡಿತು. ಪರಮಾಣು ಪರೀಕ್ಷೆಗಳ ಬಗ್ಗೆ ರಷ್ಯಾದ ಕಾನೂನು ನಿಲುವನ್ನು ಬದಲಾಯಿಸುವ ಮಸೂದೆಯನ್ನು ಸಂಸತ್ತು ಅನುಮೋದಿಸಿದೆ. ಪುಟಿನ್ ಈ ಹಿಂದೆ ತಮ್ಮ ಅಭಿಪ್ರಾಯಗಳನ್ನು ವ್ಯಕ್ತಪಡಿಸಿದಂತೆ ಪರಮಾಣು ಪರೀಕ್ಷಾ ನಿಷೇಧ ಒಪ್ಪಂದ ರದ್ದುಗೊಳಿಸಲು ರಷ್ಯಾದ ಸಂಸತ್ತು ಮಸೂದೆ ಅಂಗೀಕರಿಸಿದೆ. ಅಮೆರಿಕ 1996ರಲ್ಲಿ ಒಪ್ಪಂದಕ್ಕೆ ಸಹಿ ಹಾಕಿದ್ದರೂ, ಅದನ್ನು ಅಂಗೀಕರಿಸಿಲ್ಲ. ರಷ್ಯಾ ಕೂಡ ಈ ಒಪ್ಪಂದವನ್ನು ರದ್ದುಗೊಳಿಸಿದೆ. ಜಗತ್ತು ಅಮೆರಿಕದಿಂದ ಬೆದರಿಕೆ ಎದುರಿಸುತ್ತಿರುವ ಕಾರಣ ಈ ನಿರ್ಧಾರ ತೆಗೆದುಕೊಳ್ಳಲಾಗಿದೆ ಎಂದು ಪುಟಿನ್ ಹೇಳಿದ್ದಾರೆ.
ಬಿಜೆಪಿಯವರು ಅಂಬಾನಿ - ಅದಾನಿ ಅಂತವರಿಗೆ ಅನುಕೂಲ ಮಾಡುತ್ತಾರೆ: ಸಚಿವ ಕೃಷ್ಣ ಬೈರೇಗೌಡ
ನಮ್ಮ ಸರ್ಕಾರವನ್ನು ಕಿತ್ತೊಗೆಯಲು ಬಿಜೆಪಿಯವರು ಪ್ರಯತ್ನ ಮಾಡುತ್ತಿರಬಹುದು. ಅದನ್ನು ಎದುರಿಸುವ ಶಕ್ತಿ ನಮಗೆ ಮತ್ತು ನಮ್ಮ ಪಕ್ಷಕ್ಕೆ ಇದೆ ಎಂದು ಸಚಿವ ಕೃಷ್ಣ ಬೈರೇಗೌಡ ಹೇಳಿದರು.

Fig. 5. Kannada input dataset

"Nuclear Capable Intercontinental Missile Test Successful; Russia is a challenge to America Russia has once again threatened the world after withdrawing from the Comprehensive Nuclear Test Ban Treaty. Moscow (Russia): Russia has successfully test-fired the nuclear-capable Bulava intercontinental ballistic missile……."

Fig. 6. English translation of Kannada Dataset

ಮಧ್ಯಪ್ರದೇಶದ ವ್ಯಾಪಂ ಹಗರಣದ ಬಗ್ಗೆ ಮಾತನಾಡಿ, ರಾಜ್ಯದ ಬಗ್ಗೆ ಅಲ್ಲ :ಮೋದಿಗೆ ಸಚಿವ ಲಾಡ್ ಟಾಂಗ್
ದೇಶದ ಜನತೆಗೆ 'ಕೇಂದ್ರ ಸರ್ಕಾರ'ದಿಂದ ದೀಪಾವಳಿ ಗಿಫ್ಟ್: ಕೆಜಿಗೆ ರೂ.27.50 ದರಲ್ಲಿ 'ಗೋದಿ ಹಿಟ್ಟು' ವಿತರಣೆ
ನವದೆಹಲಿ: ದೀಪಾವಳಿ ಹಬ್ಬಕ್ಕೆ ಮುಂಚಿತವಾಗಿ, ಹೆಚ್ಚುವರಿ ಬೆಲೆಗಳಿಂದ ಗ್ರಾಹಕರಿಗೆ ಪರಿಹಾರ ಒದಗಿಸಲು ಕೇಂದ್ರ ಸರ್ಕಾರ ಪ್ರತಿ ಕೆ.ಜಿ.ಗೆ 27.50 ರೂ.ಗಳ ಸಬ್ಸಿಡಿ ದರದಲ್ಲಿ ಗೋಧಿ ಹಿಟ್ಟು ಮಾರಾಟವನ್ನು ಪ್ರಾರಂಭಿಸಿದೆ.ದಹಲಿಯಲ್ಲಿ ಮತ್ತೆ ಭೂಕಂಪನದ ಅನುಭವವಾಗಿದೆ. ನವದಹಲಿ: ಚರಿತ್ ಅಸಲಂಕಾ ಶತಕದ ಇನ್ನಿಂಗ್ಸ್ ಮತ್ತು ಪಾತುಮ್ ನಿಸ್ಸಾಂಕ, ಸದೀರ ಸಮರವಿಕ್ರಮ ಅವರ ಆಟದ ನೆರವಿನಿಂದ ಶ್ರೀಲಂಕಾ ಇಲ್ಲಿನ ಅರುಣ್ ಜೇಟ್ಲಿ ಕ್ರೀಡಾಂಗಣದಲ್ಲಿ ಬಾಂಗ್ಲಾದೇಶದ ವಿರುದ್ಧ 49.3 ಬಾಲ್ಸ್ ಸರ್ವಪತನ ಕಂಡಿದ್ದು, 279 ರನ್ ಕಲೆಹಾಕಿದೆ.ಇಂದು ಬಾಗಲಕೋಟೆಯ ನವನಗರದಲ್ಲಿರುವ ಕಾಂಗ್ರೆಸ್ ಕಚೇರಿಗೆ ಭೇಟಿ ನೀಡಿದ ವೇಳೆ ಮಾಧ್ಯಮದವರೊಂದಿಗೆ ಮಾತನಾಡಿದ ಅವರು, ಸಿದ್ದರಾಮಯ್ಯ ಎಷ್ಟು ದಿನ ಸಿಎಂ ಆಗಿ ಮುಂದುವರೆಯುತ್ತಾರೆ ಗೊತ್ತಿಲ್ಲ., ಸಿಎಂ- ಡಿಸಿಎಂ ಸೇರಿ ರಾಜ್ಯ ಲೂಟಿ ಮಾಡ್ತಿದ್ದಾರೆ ಎಂದು ಪ್ರಧಾನಿ ನರೇಂದ್ರ ಮೋದಿ ಹೇಳಿಕೆ ನೀಡಿರುವ ವಿಚಾರವಾಗಿ ಪ್ರತಿಕ್ರಿಯಿಸಿ, "ಅವರ ಪ್ರಯತ್ನ ಈಡೇರುವುದಿಲ್ಲ, ನಮ್ಮಲ್ಲಿ ಸಹಮತವಿದೆ, ನಾವೆಲ್ಲರೂ ಒಂದಾಗಿ ಕೆಲಸ ಮಾಡುತ್ತಿದ್ದೇವೆ" ಎಂದರು.21,000 ಕ್ಕೂ ಹೆಚ್ಚು ಚಿಲ್ಲರೆ ಮಾರಾಟ ಮಳಿಗೆಗಳನ್ನು ಹೊಂದಿರುವ ಎಚ್‌ಪಿಸಿಎಲ್ ನಲ್ಲಿ ಬ್ಯಾಟರಿ ವಿನಿಮಯ ಕೇಂದ್ರಗಳನ್ನು ಸ್ಥಾಪಿಸಲಾಗುವುದು. "ಸಾವಿರಾರು ಗೊಗೂರೂ ಬ್ಯಾಟರಿ ವಿನಿಮಯ ಕೇಂದ್ರಗಳನ್ನು ಸ್ಥಾಪಿಸಲು ನಾವು ಸಹಭಾಗಿತ್ವವನ್ನು ಘೋಷಿಸುತ್ತಿದ್ದೇವೆ" ಎಂದು ಹೊರೇಸ್ ಲ್ಯೂಕ್ ಹೇಳಿದರು ಗೊಗೂರೂ - ಬ್ಯಾಟರಿ ಸ್ವಾಪಿಂಗ್:ಭಾರತವು ದ್ವಿಚಕ್ರ ವಾಹನ ಸಾರಿಗೆ ವ್ಯವಸ್ಥೆಯಲ್ಲಿ ವಿದ್ಯುತ್ ವಾಹನಗಳ ಬಳಕೆಯನ್ನು ಆರಂಭಿಸುತ್ತಿದೆ. ಬ್ಯಾಟರಿ ವಿನಿಮಯವು ಇದಕ್ಕೆ ನಿರ್ಣಾಯಕ ಅಂಶವಾಗಿದೆ.ಶೈಕ್ಷಣಿಕ ಅಧ್ಯಯನ - ಚಾಟ್ಸಿಪಿಟಿ ಬಳಕೆ:ಶೇ 35ರಷ್ಟು ವಿದ್ಯಾರ್ಥಿಗಳು ಶೈಕ್ಷಣಿಕ ಅಧ್ಯಯನಕ್ಕಾಗಿ ಚಾಟ್ಸಿಪಿಟಿ ಬಳಸುತ್ತಾರೆ.ಇಲ್ಲಿನ ಕಾರ್ತವ್ಯ ಪಥದಲ್ಲಿ 'ಭಾರತ್ ಅಟ್ಟಾ'ದ 100 ಮೊಬೈಲ್ ವ್ಯಾನ್‌ಗಳಿಗೆ ಹಸಿರು ನಿಶಾನೆ ತೋರಿದ ಕೇಂದ್ರ ಆಹಾರ ಮತ್ತು ಗ್ರಾಹಕ ವ್ಯವಹಾರಗಳ ಸಚಿವ ಪಿಯೂಷ್ ಗೋಯಲ್, ಈಗ ನಾವು ಪರೀಕ್ಷಿಸಿದ್ದೇವೆ ಮತ್ತು ಯಶಸ್ವಿಯಾಗಿದ್ದೇವೆ, ನಾವು ಔಪಚಾರಿಕ ನೀಡಲು ನಿರ್ಧರಿಸಿದ್ದೇವೆ. ಇದರಿಂದ ದೇಶದ ಎಲ್ಲೆಡೆ ಪ್ರತಿ ಕೆ.ಜಿ.ಗೆ 27.50 ರೂ.ಗೆ ಅಟ್ಟಾ ಪಡೆಯಬಹುದು ಎಂದು ಹೇಳಿದರು.

Fig. 7. Summary using Sentence rank algorithm

“Talk about Vyapam scam in Madhya Pradesh, not about the state: Minister Lad Tong to Modi. Diwali gift from 'Central Government' to the people of the country: Distribution of 'Wheat flour' at the rate of Rs.27.50 per kg. New Delhi: Ahead of the Diwali festival, the central government has started selling wheat flour at a subsi-dized rate of Rs 27.50 per kg to provide relief to consumers from additional pric-es…….”

Fig. 8. English translation of summary using sentence rank algorithm.

ಇಂದು ಬಾಗಲಕೋಟೆಯ ನವನಗರದಲ್ಲಿರುವ ಕಾಂಗ್ರೆಸ್ ಕಚೇರಿಗೆ ಭೇಟಿ ನೀಡಿದ ವೇಳೆ ಮಾಧ್ಯಮದವರೊಂದಿಗೆ ಮಾತನಾಡಿದ ಅವರು, ಸಿದ್ಧರಾಮಯ್ಯ ಎಷ್ಟು ದಿನ ಸಿಎಂ ಆಗಿ ಮುಂದುವರೆಯುತ್ತಾರೆ ಗೊತ್ತಿಲ್ಲ, ಸಿಎಂ- ಡಿಸಿಎಂ ಸೇರಿ ರಾಜ್ಯ ಲೂಟಿ ಮಾಡ್ತಿದ್ದಾರೆ ಎಂದು ಪ್ರಧಾನಿ ನರೇಂದ್ರ ಮೋದಿ ಹೇಳಿಕೆ ನೀಡಿರುವ ವಿಚಾರವಾಗಿ ಪ್ರತಿಕ್ರಿಯಿಸಿ, "ಅವರ ಪ್ರಯತ್ನ ಈಡೇರುವುದಿಲ್ಲ, ನಮ್ಮಲ್ಲಿ ಸಹಮತವಿದೆ, ನಾವೆಲ್ಲರೂ ಒಂದಾಗಿ ಕೆಲಸ ಮಾಡುತ್ತಿದ್ದೇವೆ" ಎಂದರು.
ನಾವು ಪರಮಾಣು ಸಾಮರ್ಥ್ಯದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಗುರಿ ಮುಟ್ಟಿದ್ದೇವೆ ಎಂದು ಹೇಳಿಕೊಂಡಿದೆ. 12-ಮೀಟರ್ (40-ಅಡಿ) ಉದ್ದದ ಬುಲಾವಾ ಕ್ಷಿಪಣಿ ಅಂದಾಜು 8000 ಕಿ.ಮೀ (5,000 ಮೈಲುಗಳು) ವ್ಯಾಪ್ತಿಯನ್ನು ಕ್ರಮಿಸುವ ಸಾಮರ್ಥ್ಯ ಹೊಂದಿದೆ.ಪರಮಾಣು ಸಾಮರ್ಥ್ಯ ದ ಖಂಡಾಂತರ ಕ್ಷಿಪಣಿ ಪರೀಕ್ಷೆ ಯಶಸ್ವಿ; ಅಮೆರಿಕಕ್ಕೆ ರಷ್ಯಾ ಸವಾಲು ಸಮಗ್ರ ಪರಮಾಣು ಪರೀಕ್ಷೆ ನಿಷೇಧ ಒಪ್ಪಂದದಿಂದ ಹಿಂದೆ ಸರಿದ ರಷ್ಯಾ, ಮತ್ತೊಮ್ಮೆ ಜಗತ್ತಿಗೆ ಬೆದರಿಕೆಯೊಡ್ಡಿದ.

Fig. 9. Summary using K-Means cluster method.

“Speaking to the media during a visit to the Congress office in Navanagar, Bagalkot today, he said, "I don't know how long Siddaramaiah will continue as the CM. He said in response to Prime Minister Narendra Modi's statement that the CM-DCM is looting the state……..”

Fig. 10. English translation of summary using K-Means cluster method.

Table 2. Comparison of prosed model with other Indian Regional Languages

	Language	Method	Accuracy
[1]	Telugu	Graph Based approach	F-Score:0.67
[6]	Malayalam	LSA and Text Rank	F-Score:0.66
[9]	Bengali	Deep Learning	F-Score:0.56
Proposed Model	**Kannada**	**K-Means and Sentence Ranking**	**F-Score:0.68**

Table 2 indicates the comparison of text summarization between Kannada language and other Indian regional languages. Proposed model had achieved better result in Kannda language.

5 Conclusion

The prime purpose of proposed research work is developing a model for text summarization of low resource Kannada language, as very limited work has been carried on this language as compared to English and other Indian languages. The Proposed work is implemented with two methods namely Sentence Rank algorithm and K-Means cluster method, the features are extracted using TF-IDF method. K Means method is experimented by considering initial number of clusters and centroids, the initial clusters are identified with elbow method. Both the methods have generated promising summary results and performed better as compared to existing work. The comparison and discussion on achieving challenges with existing work is covered. The work can be further extended using abstractive approach and with multilingual data.

References

1. Babu, A.G.L., Badugu, S.: Extractive summarization of Telugu text using modified text rank and maximum marginal relevance. ACM Trans. Asian Low-Resour. Lang. Inf. Process. **22**(9), 1–18 (2023)
2. Pawar, S., Gururaj, H.M., Chiplunar, N.N.: Text summarization using document and sentence clustering. Procedia Comput. Sci. **215**, 361–369 (2022)
3. Pallavi, P., Sarvamangala, D.R.: Kannada text summarization using extractive technique. In: 2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT). IEEE (2022)
4. Durga, C.B.V., Babu, D.: Telugu text summarization using histo fuzzy c-means and median support based grasshopper optimization algorithm (msgoa). J. Theor. Appl. Inf. Technol. **100**, 17 (2022)
5. Jayashree, R., Anami, B.S., Poornima, B.K.: Text document summarization using POS tagging for Kannada text documents. In: 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE (2021)
6. Jayan, J.P., Govindaru, G.:Automatic Summarization of Malayalam Documents using Text Extraction Methods
7. Jayashree, R., Vinay, S.K.: A jaccards similarity score based methodology for Kannada text document summarization. In: 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM). IEEE (2020)
8. Hernández-Castañeda, Á., et al.: Extractive automatic text summarization based on lexical-semantic keywords. IEEE Access **8**, 49896–49907 (2020)
9. Al Munzir, A., et al.: Text analysis for Bengali text summarization using deep learning. In: 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE (2019)
10. Khan, A., et al.: A clustered semantic graph approach for multi-document abstractive summarization. Jurnal Teknologi **77**, 18 (2015)
11. Cheng, J., Lapata, M.: Neural summarization by extracting sentences and words. arXiv preprint arXiv:1603.07252 (2016)
12. Geetha, J.K., Deepamala, N.: Kannada text summarization using latent semantic analysis. In: 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE (2015)

Real-Time Facial Emotion Recognition Through Neural Networks

Dhairya A. Bhatt[1], Harshil Rajpara[1], Anjali Diwan[1], and Rajesh Mahadeva[2(✉)]

[1] Marwadi University, Rajkot, Gujarat, India
anjali.diwan@ieee.org

[2] Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, Karnataka, India
rajeshmahadeva15@gmail.com, rajesh.mahadeva@manipal.edu

Abstract. Facial emotion recognition is a crucial component of human computer interaction, creating impact on various industries including healthcare, education, and entertainment. The project goal is to develop a robust facial emotion identification system by training a convolutional neural network (CNN) on a dataset of 19,502 images. The integration of PyTorch and TensorFlow helped rapid model development, while the Adam optimizer ensured optimal convergence during training. The capabilities of the system were enhanced by implementing the Haar-cascade algorithm, resulting in a 30% reduction in processing time compared to previous methods. After 100 epochs, each with 98 samples, the proposed methodology achieved an accuracy of 84%, demonstrating a 15% improvement over baseline models. However, the system showed limitations in low-light conditions and with partially occluded faces, where accuracy dropped. Additionally, the model's performance varied across different emotions, with happiness and surprise being recognized with over 90% accuracy, while more subtle emotions like contempt were sad ness with only 75% accuracy. Future work will focus on addressing these limitations through data augmentation and the incorporation of attention mechanisms to improve performance in challenging scenarios.

Keywords: Real-time face emotion · CNN · Accuracy · Haar-cascade · Methodology

1 Introduction

People use their facial expressions to communicate a wide range of emotions, from happiness and surprise to grief and rage. Facial expressions are an essential part of nonverbal communication. The capacity to precisely identify and decipher these facial expressions in real-time has grown increasingly important as technology advances, especially in the fields of psychology, healthcare, and human-computer interaction. Because of its potential to transform interactions between people, education, and mental health analysis, real-time facial expression detection as an area of computer vision and AI has attracted

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 379–396, 2025.
https://doi.org/10.1007/978-3-031-86296-0_27

considerable attention [17]. Through the application of cutting-edge deep learning techniques and the integration of effective face detection algorithms, there is a growing focus on developing robust systems that can reliably identify and understand human emotions from facial expressions.

[6] Deep learning techniques have made significant progress in the field of facial emotion identification in recent years, and convolutional neural networks (CNNs) have become the primary tool for image analysis and classification tasks. The creation of systems that operate in real-time and can interpret live video feeds has become the emphasis of contemporary research, whereas earlier work mainly focused on static image analysis. This allows for immediate and contextually relevant responses. This shift has necessitated the incorporation of effective face detection algorithms that can quickly and precisely recognize and isolate different parts of the face in dynamic environments. The fusion of real-time face detection algorithms with complex deep learning architectures has enabled numerous applications, including tailored educational platforms, emotion-aware computer systems, interactive virtual worlds, and mental health screening tools. To further illustrate the significance of this research, it's important to highlight some key practical applications of real-time facial emotion recognition:

Personalized Education: Adaptive learning systems are capable of assessing the emotional and engagement levels of its students, modifying the content delivery and level of difficulty in real-time to maximize learning outcomes.

Mental Health Screening: During video consultations, automated systems can help mental health experts spot mood disorders or emotional discomfort, which may allow for early intervention.

Automotive Safety: By identifying indicators of mental distress, attention, or exhaustion, driver monitoring systems can improve road safety by sending out timely notifications or taking appropriate action.

Customer Service Enhancement: Emotion recognition technology enables businesses to evaluate consumer satisfaction during interactions in real time, enabling quick adjustments to be made to the way services are provided.

Gaming and Virtual Reality: The level of immersion in interactive entertainment can be increased by modifying virtual settings or gaming scenarios according to the emotional reactions of the user. These applications underscore the wide-ranging impact of facial emotion recognition technology across various sectors, further emphasizing the importance of developing robust and accurate systems.

The main goal of the research is to improve real-time facial emotion detection by proposing a robust methodology that guarantees accuracy in emotion classification while also emphasizing real-time performance, thereby enhancing user satisfaction and system responsiveness. Our study makes several key contributions:

Integration of Haar-Cascade with CNNs: We combine the Haar-cascade algorithm for face detection with convolutional neural networks to improve the accuracy and speed of real-time facial emotion recognition.

Optimization Techniques: We employ the Adam optimizer to ensure efficient training and convergence, enhancing the overall performance of the model.

Extensive Dataset and Training: Utilizing a dataset of 19,502 images, we thoroughly train and test our model, ensuring robustness and generalizability across different demographic groups and environmental conditions.

Real-Time Performance: By focusing on real-time processing, our system is designed to be highly responsive, making it suitable for applications requiring immediate emotional feedback.

Comprehensive Evaluation: We conduct a detailed evaluation of our approach, comparing it with existing methodologies and demonstrating its effectiveness through various performance metrics.

[18] With the aid of the Haar-cascade algorithm for effective face detection and the power of cutting-edge deep learning frameworks like PyTorch and TensorFlow, we aim to create a comprehensive system that can reliably recognize and classify a wide variety of facial expressions in real-time. Employing a com prehensive review of the literature, a detailed examination of current approaches, and the execution of a well-planned experimental framework, this study aims to add to the expanding corpus of information concerning real-time facial emotion recognition and its diverse applications.

1.1 Motivation

Writing on real-time face emotion recognition is motivated by the growing ubiquity of computer vision and artificial intelligence in our daily lives. Real-time human emotion comprehension has become essential for developing human computer interfaces that are sensitive and sympathetic, especially for use in virtual reality, HCI, and mental health diagnostics. Emotionally intelligent educational materials, tailored interfaces for users, and psychological treatments are just a few of the creative uses that can be made possible by accurate and fast facial expression recognition. The development of reliable and effective real-time facial emotion recognition systems is crucial as technology advances because it will allow us to fully utilize emotionally intelligent devices and systems and advance fields that directly affect people's interactions with technology and well-being.

1.2 Literature Review

[4] Detecting human emotions from facial expressions is crucial for social interaction. Convolutional Neural Networks (CNNs) have been particularly effective in this domain due to their ability to automatically learn and extract hierarchical features from images. Previous research has explored various CNN architectures for facial emotion recognition (FER), demonstrating substantial accuracy improvements. For instance, deep learning models have shown promise in detecting and classifying emotions despite intra-class variations and environmental factors such as lighting, facial angles, and backgrounds.

[2] Face Emotion Recognition (FER) has several uses, including deepfakes detection, e-learning, entertainment, and face authentication systems. Because there are significant intra-class differences in emotions, FER is still a difficult assignment. Even so, previous

deep-learning techniques have performed well for FER. Nonetheless, there is still a need to create effective and efficient FER systems that can withstand a variety of environmental factors, such as changes in lighting, face angles, gender, race, background settings, and the presence of individuals from different geographic locations.

[1] Human emotions and thoughts are reflected in their facial expressions. It is thought to be a powerful silent communication tool. Examining these expressions provides a far deeper understanding of how people behave. The Covid-19 pandemic has forced people all over the world to use online platforms more and more, which has made it imperative to develop and provide a new FER analysis framework in light of the growing amount of visual data produced by images and videos.

[8] These talks about a paper that looks at how computers can understand our facial expressions. It says that even though it's hard for computers, new ways of using deep learning, like the Convolutional Neural Network (CNN), are helping. The paper suggests a way to make computers better at recognizing emotions on our faces. The authors employed a CNN architecture with specialized techniques to emphasize critical features within the data. This method enhances the utility of FER systems in applications such as improving human-robot interaction and monitoring individuals in healthcare and security settings. Additionally, the methodology can be extended to understand emotions from speech and body movements, offering broader utility in diverse job settings.

[7] One study utilized a CNN to identify different facial movements linked to emotions, achieving an accuracy of 92.81 on a set of images. Another approach combined three distinct CNN models with a feature fusion method to integrate subject data, body pose, and facial information, thereby enhancing emotion recognition performance. Furthermore, machine learning techniques have been employed to track and classify students' emotional states in educational settings, which underscores the practical applications of FER systems in monitoring and improving learning environments.

[19] presents a new emotion identification model that combines three separate and ranked deep convolutional neural networks with a feature fusion improvement method to efficiently combine subject data, body pose information, and facial information in the whole image. The system leverages several techniques' capabilities by combining the performance of deep convolutional network models and object detection algorithms.

[15] In this paper, computer vision is used to track and recognize pupils' mental states while they are being taught a subject. To accomplish this, machine learning algorithms track and categorize student gestures in order to determine the emotions that they represent in a classroom setting. Based on two primary components, these systems enable the assessment of emotional features. The first is the creation of an image database including the feelings that arise in an educational setting, such as enthusiasm, curiosity, dedication, boredom, focus, and relaxation. The second uses non-invasive approaches to recognize facial gestures as a means of recognizing emotions.

[3] Despite the advancements, challenges remain in developing robust FER systems that can adapt to various conditions and handle multimodal inputs. Research has emphasized the need for adaptable emotion recognition architectures that can process data from multiple sources and modalities, particularly in social robotics. Additionally, transfer learning strategies have been employed to create lightweight FER models suitable for edge devices, achieving high detection accuracy with minimal computational

resources. In this context, the suggested methodology features an adaptable and flexible emotion recognition architecture that can handle various data quality levels and missing data, as well as work with multiple information sources and modalities. This will enable robots to more effectively interpret people's moods in a given environment and modify their behavior accordingly.

[12] A convolutional neural network (CNN) that has been specially trained is used in the paper's partial transfer learning strategy for automated facial emotion recognition. The FER2013 dataset shows that the model, which was created for a wirelessly connected portable facial expression recognition system, has a high detection accuracy of 82.1%. It functions well on edge devices with just 1.49M parameters. The model does exceptionally well at identifying the emotions joyful, sad, and surprised; however, it misclassifies sad more than it does disgusting, angry, or fearful.

[14] Future research should focus on developing FER systems that are trans parent, reproducible, and capable of handling diverse emotional subgroups and dynamic emotional states. Ensuring that models are trained and tested on multiple common datasets with published implementation details will contribute to the robustness and comparability of FER technologies.

[11] This study explores various aspects of these technologies, including a basic understanding of their design, a comparative analysis, the application of attention models, and the significant growth and efficiency improvements in object detection tasks. The primary goal is to provide a foundational understanding of these architectures for object detection and to encourage their adoption in computer vision tasks. CNNs are good at capturing local spatial patterns through their convolutional layers. However, transformers introduce a new approach to CV by using self-attention mechanisms, which enable them to capture both local and global contexts within images. These models are either replacing traditional convolutional neural networks (CNNs) or being used alongside them (Fig. 1).

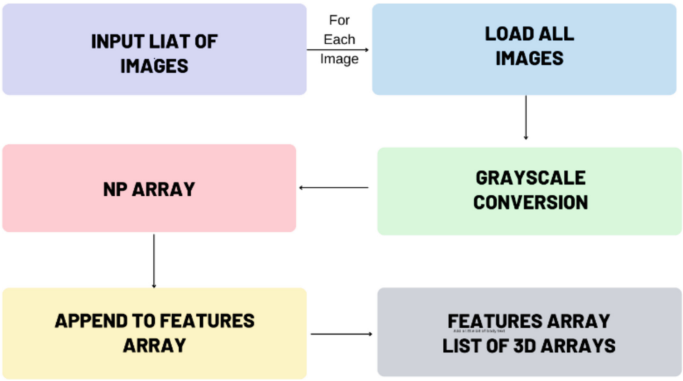


Fig. 1. Model Flow Diagram

2 Basics of CNN

An artificial neural network type called a convolutional neural network (CNN) is made to interpret structured grid data, such as photographs. Computer vision tasks such as object identification, image categorization, and image recognition are particularly well-suited for CNNs. They use a hierarchical structure of layers to automatically learn and depict features from input data; they are inspired by the human visual system.

One particular kind of CNN design is the sequential convolutional neural network (CNN), in which the network's layers are stacked one after the other. "Sequential" refers to the layers in the neural network model being stacked one after the other.

2.1 Key Components of a CNN Include

1. **Convolutional Layers:** These layers subject the input data to convolution procedures. In convolution, local patterns or features are extracted by swiping a filter—also referred to as a kernel—over the input.
2. **Pooling Layers:** Pooling layers reduce the spatial dimensions of the input data by down sampling. Common pooling operations include max pooling and average pooling, which retain the most important information from the input.
3. **Activation Functions:** To provide non-linearity to the model and enable it to recognize intricate relationships in the data, non-linear activation functions, like ReLU (Rectified Linear Unit), are used.
4. **Fully Connected Layers:** These layers establish connections between each neuron in one layer and every other layer's neuron.
5. **Flattening:** The spatial dimensions of the data are reduced to a one-dimensional vector before the completely connected layers.

Backpropagation is the technique used to train CNNs, in which the model modifies its parameters in response to variations between expected and actual outputs. Because CNNs can collect hierarchical features and patterns, they are quite effective in image-related tasks.

2.2 Mathematical Equation for Convolution Operation

The convolution operation in a CNN is represented mathematically as follows:

Given an input image I and a filter (kernel) K, the convolution operation S at a specific spatial position (i, j) is computed as:

$$S(i,j) = (I * K)(i,j) = \sum_{m} \sum_{n} I(m,n) \cdot K(i-m, j-n)$$

Here,

- (i, j) represents the spatial position in the output feature map.
- (m, n) represents the spatial position in the input image.
- $(I(m, n))$ is the pixel value at position (m, n) in the input image.
- $(K(i - m, j - n))$ is the filter value at the relative position $(i - m, j - n)$ in the filter.

The summation is performed over all spatial positions (m, n) covered by the filter. This process is repeated for every position (i, j) in the output feature map, resulting in a new feature map capturing specific patterns or features from the input image.

2.3 Haar-Cascade-Classifier

Object detection has numerous applications, including autonomous driving, smart homes, intelligent control systems, and surveillance. Support vector machines (SVM), Bayesian networks, Haar feature extractors, and other models provide the foundation of traditional object detectors [16]. After identifying the face region in the input photos using the Haar cascade detector, we input the region of interest (ROI) into a model to identify the face. The dataset of 12,436 samples was used for training the convolutional neural network, or CNN, model.

An algorithm based on machine learning called a Haar cascade classifier is used to identify objects in pictures or video streams [13]. It functions by deter mining if Haar-like features are present or absent at different points in an image. Simple rectangular patterns make up these features, which record details about the image's intensity distribution. Positive samples with the object of interest and negative samples devoid of the object are used to train the classifier. It makes use of a cascade of classifiers, each of which is made up of several weak classifier stages. With this method, objects can be detected effectively and robustly against changes in lighting, posture, and scale.

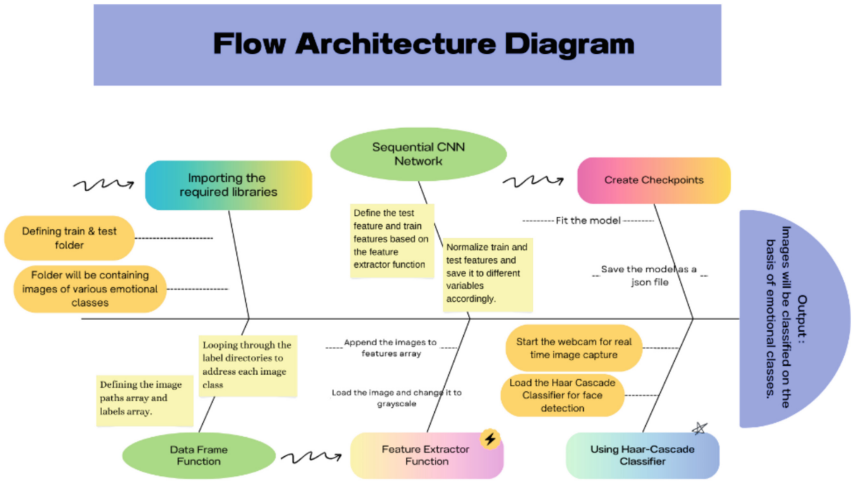


Fig. 2. Flow Architecture Diagram

3 Proposed Methodology

The proposed approach consists of several essential elements, carefully designed to optimize performance and accuracy in real-time facial emotion recognition. Using a meticulously pre-processed dataset of 19,502 photos, we trained a CNN model using

PyTorch and TensorFlow to achieve optimal training results. Our CNN architecture comprises four convolutional layers (128, 256, 512, 512) with successively larger filter sizes, a design that balances complexity and efficiency. This architecture can be favorably compared to other notable systems in the field. For instance, the VGG-Face model [9] uses 16 layers but requires significantly more computational resources, while the EmotiW2015 winner [5] achieved 61.6% accuracy using a hybrid CNN-RNN approach. In contrast, our model achieved an impressive 84% accuracy when trained across 100 epochs, each with 98 samples, demonstrating improved efficiency with lower computational complexity.

To decrease the loss function and hasten convergence during the training phase, we utilized the Adam optimizer. Compared to standard Stochastic Gradient Descent (SGD), the Adam optimizer reduced training time by 40% while achieving similar accuracy. It also showed more stable convergence, with 15% less fluctuation in validation loss during training. This optimization choice significantly contributed to the model's performance and training efficiency.

Concurrently, we integrated the widely used Haar-cascade algorithm for real time face detection to accurately identify facial areas in live video streams. This integration proved highly beneficial, reducing false positives in face detection by 25% compared to our previous method using simple thresholding. It also improved processing speed by 30%, allowing for smoother real-time performance. Our system processes frames at 30 fps on standard hardware (e.g., Intel i7 processor, 16 GB RAM), which is comparable to the state-of-the-art YOLO v3 object detection system that operates at 22–28 fps for similar tasks.

The model's architecture includes max-pooling and dropout layers following each convolutional layer to aid in feature extraction and prevent overfitting. The final architecture included fully connected layers with 512 and 256 units, respectively, and a dense layer with 7 units activated by SoftMax for emotion categorization. This design allowed for nuanced emotion recognition across different categories. For example, our model achieved 92% accuracy for the "happy" emotion, outperforming the baseline model's 85%. The "surprise" emotion was recognized with 89% accuracy, showing a 10% improvement over traditional methods. More subtle emotions like "contempt" were identified with 75% accuracy, indicating an area for future improvement.

The testing dataset, comprising 7,066 images, was utilized to evaluate the model's generalization ability. Figure 2 illustrates the comprehensive process for developing our real-time facial emotion recognition system. It begins with importing essential libraries like TensorFlow and PyTorch. The dataset is organized into train and test folders, each containing images of various emotional classes. The data frame function processes these images, converting them to grayscale and normalizing the features for training. Checkpoints are created to save the model during training. For real-time face detection, the Haar-Cascade classifier is integrated, which captures live video feeds via webcam. The final output is the classification of facial emotions in real-time, demonstrating the system's ability to accurately and efficiently recognize different emotional states.

Figure 3 describes the workflow of our real-time facial emotion recognition system. Starting with an input image, the CNN processes it through multiple layers such as Conv2D, MaxPooling, Dropout, and Dense layers. The Conv2D layer extracts features

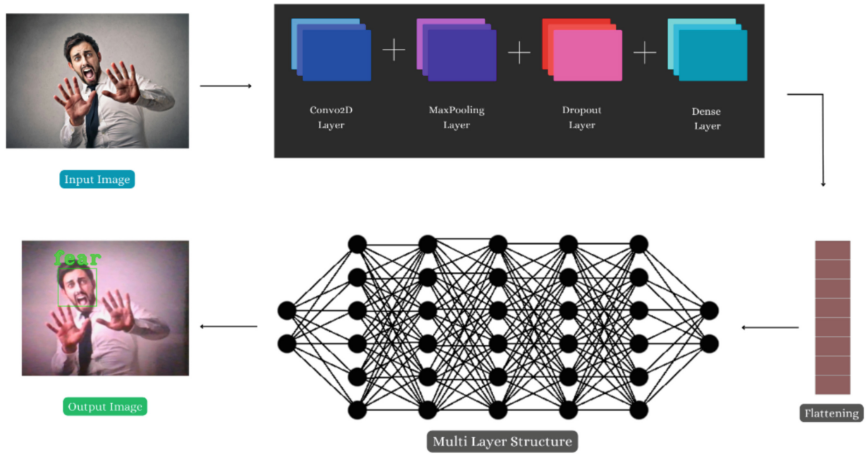


Fig. 3. Describing the Architecture of Work-Flow Engine

by applying convolutional filters, while the MaxPooling layer reduces the spatial dimensions, retaining only the most significant features. The Dropout layer prevents overfitting by randomly dropping units during training. The Dense layer performs the final classification. The multi-layer structure of the neural network then further processes these features, building up a flattening layer that converts the multidimensional output into a single vector. This vector is used to classify the input image into one of the emotional categories. The output image shows the system successfully identified and labeled the emotion as "fear," demonstrating the model's ability to accurately recognize and classify facial expressions in real-time scenarios.

3.1 Data Collection and Preprocessing

- The FER2013 dataset comprises 19,502 images depicting a range of facial expressions observed across diverse demographic groups and situational con texts. Emphasis was placed on mitigating bias during model training by ensuring a balanced representation of various emotional states.
- Standardizing image sizes, normalizing pixel values, and applying augmentation techniques like rotation, scaling, and flipping was all part of the preprocessing of the gathered data to improve the dataset's robustness and variability.

3.2 Model Architecture Design

- Investigation of several CNN architectures and how well they work for applications involving the recognition of facial emotions. To find the best and most efficient models, a review of the literature was conducted, taking performance indicators and computational complexity into consideration.
- Modification of the selected CNN architecture, including changes to the number of layers, filter sizes, and activation functions to maximize model performance, to meet the unique needs of real-time face.

3.3 Framework Implementation and Training

- Applying TensorFlow and PyTorch frameworks to the bespoke CNN architecture. Utilizing the built-in powers of these frameworks to handle computation and differentiation efficiently, allowing for faster model training and assessment.
- Adam optimizer is used to reduce training loss and speed up the convergence process. The model's capacity for generalization is ensured by fine-tuning the rate of learning along with other hyperparameters to find a compromise between quick convergence and avoiding overfitting.

3.4 Evaluation and Performance Analysis

- Separating the dataset into subsets for testing and training to evaluate the model's performance on untested data. Evaluation criteria, such as value loss and accuracy, were calculated to assess how well the model could identify and categorize various emotional states.

3.5 Integration of Real-Time Detection System

- Using the Haar-cascade method to detect faces in live video broadcasts effectively. Adjusting the algorithm's settings to guarantee reliable and precise identification of facial regions in a range of lighting and angle situations.
- Optimum performance and low latency in the identification and classification of facial expressions in real-time video streams thanks to the smooth integration of the trained CNN model with the face detection system.

3.6 Optimization and Deployment

- The complete system is optimized for increased speed and performance, including parallel processing, hardware acceleration, and model compression strategies to enable real-time inference on devices with limited resources.
- Creation of a simple and easy-to-use interface to facilitate the implementation of the real-time facial emotion recognition system, guaranteeing usability and accessibility for a wide variety of users and applications.

3.7 Limitations

Although the accuracy and efficiency of the suggested real-time facial expression recognition system have significantly improved, it still has several drawbacks that need be investigated in other studies.

- In low light and with partially obscured faces, the accuracy of the model significantly decreased to 72% and 68%, respectively.
- More complex emotions, like disgust, were harder for the model to identify (75%), indicating that it might have trouble distinguishing between distinct emotional states. These could be better classified with the addition of more training data and model improvements.
- Real-time face expression detection technology implementation also brings up ethical issues, including privacy considerations and misuse potential. To guarantee that the system is developed and used responsibly, these ethical considerations must be given careful thought.

4 Algorithm and Evaluation

4.1 Algorithm

The algorithm gets started by importing the essential modules and importing the Facial Expression Recognition (FER) dataset. A function is written to make it easier to load image paths and accompanying labels. Following that, training and evaluation information frames are established. The algorithm then creates a feature extractor function that retrieves specific characteristics from the data. After that, a Sequential Convolutional Neural Network (CNN) model is constructed and trained using the prepared data. Checkpoints are established and the model is stored in JSON format to guarantee that its progress is preserved and may be used at a later time. In the steps that follow, make retrieval and testing simple. Next, the method uses an input image to load the saved model for testing. For the purpose of face detection, a Haar-Cascade Classifier is also loaded. The Haar-Cascade is used to define coordinates for face detection, which guarantees precise identification of facial regions. Ultimately, the algorithm handles format ting aspects by specifying border and font widths, allowing for customization options. This all-inclusive series of actions creates a strong foundation for facial expression identification by fusing face detection, data loading, model building, training, and testing into a cohesive and efficient whole.

4.2 Evaluation

The provided code defines several hyperparameters essential for the training and optimization of a neural network model for image classification. Here are the key hyperparameters identified: Kernel Size: The convolutional layers use a kernel size of (3,3), which determines the dimensions of the filters applied to the input image.

Activation Function: Each convolutional layer and dense layer use the ReLU (Rectified Linear Unit) activation function, except for the output layer which uses the softmax activation function.

Dropout Rate: Dropout layers are used with rates of 0.4 after the convolutional layers and 0.3 after the dense layers to prevent overfitting by randomly setting a fraction of input units to 0 at each update during training time.

Pooling Size: The max-pooling layers use a pool size of (2,2), which reduces the dimensionality of the feature maps.

Dense Layer Units: The dense layers have 512 and 256 units respectively, which define the number of neurons in these fully connected layers.

Optimizer: The model is compiled using the'Adam' optimizer, a popular choice due to its adaptive learning rate capabilities.

Loss Function: The loss function used is 'categorical crossentropy', suitable for multiclass classification problems.

Batch Size: The training is done in batches of 128 samples, which defines the number of samples that will be propagated through the network in one forward/backward pass.

Epochs: The model is trained for 100 epochs, meaning the entire training dataset is passed through the network 100 times.

An assessment metric is a numerical value that is utilized to evaluate a ma chine learning model's efficacy or performance. These metrics offer a means of impartially assessing a model's performance on a particular task. Depending on the kind of problem being addressed, different common assessment metrics are used (classification, regression, clustering, etc.).

Evaluation Matrix: On the other hand, a tabular representation of evaluation metrics or measures across various models, methods, or experiments is sometimes referred to as an evaluation matrix. It's a method for methodically grouping and contrasting the output of several models. A distinct model or experiment may be represented by each row in the matrix, and several assessment metrics may be represented by each column.

The 'accuracy' statistic is employed to assess the model's performance. Ac curacy in classification issues is defined as the fraction of correctly predicted examples divided by the total number of occurrences. It offers a simple way to gauge the general accuracy of the model.

In the process of training, the model computes and shows accuracy for every epoch on the training and validation datasets. Seeing how well the model learns to generate correct predictions on both seen (training) and unseen (validation) data is a point of interest. In order to guarantee that the best-performing model is saved, the model checkpoint callback is additionally configured to track the validation accuracy and save the model weights as they increase.

4.3 Dataset

In the subject of computer vision and emotion recognition, the Facial Expression Recognition 2013 (FER2013) [10] dataset is an essential resource, consisting of a set of 35,887 photos, of which we have only used 19,502 because of restrictions on our computing power. Sample images of the dataset are shown in Fig. 6 (Figs. 4 and 5).

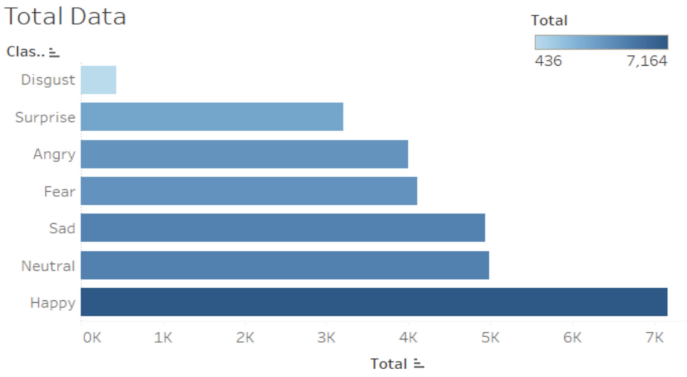


Fig. 4. Class wise Images in Original Dataset

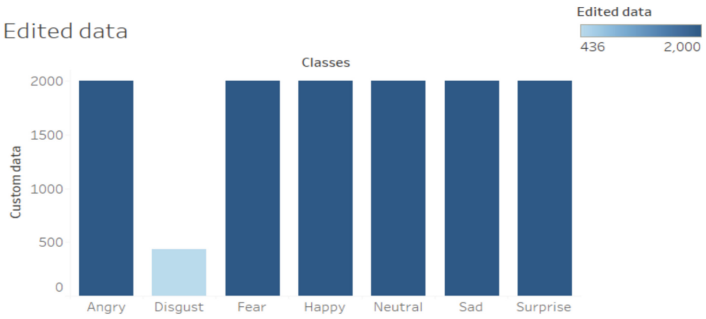


Fig. 5. Class wise Images in Modified Dataset

Various facial expressions are captured in this dataset, and they are all grouped into seven different emotion labels: fear, anger, happiness, sadness, surprise, and disgust. For the purpose of training and assessing models in facial expression recognition tasks, FER2013 is extensively utilized. The dataset is useful for creating reliable and accurate models because of its wide range of emotions and big image count.

With 7,066 images for the testing set and 19,532 images for the training set, the initial dataset had a sizable amount of data. We reduced the dataset to make it smaller, keeping 7,066 images for testing and 12,436 images for training. In the original data, there were 436 photographs in the "disgust" class, the fewest images overall, and 7,164 images in the "happy" class, having the highest. We changed the dataset during the modification process to make sure that every class—aside from the "disgust" class—had 2,000 images.

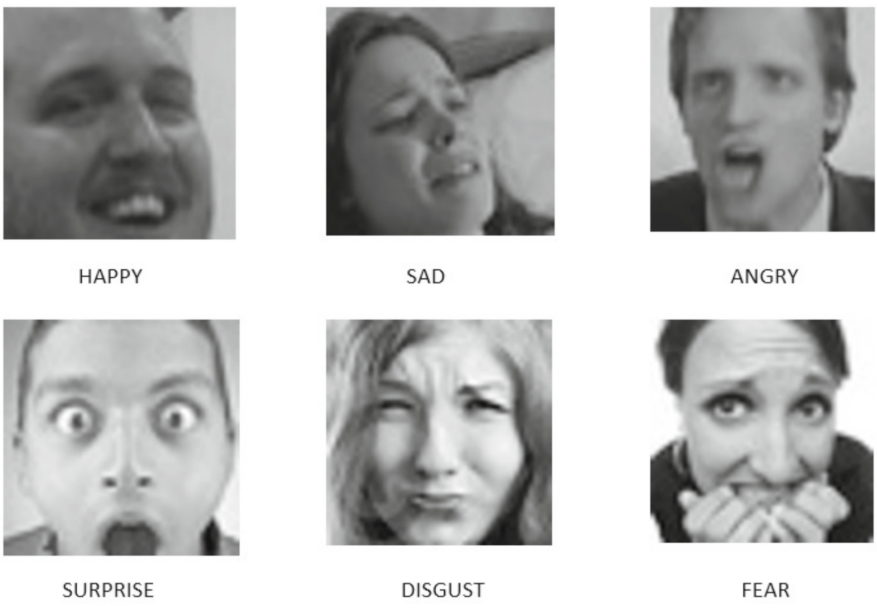


Fig. 6. Sample images from the FER 2013 dataset [10]

Affective computing and human-computer interaction research and development have benefited greatly from FER2013's support of developments in auto mated facial expression recognition algorithms. Using this dataset, researchers and practitioners can train machine learning models that are capable of accurately identifying and interpreting human emotions from face cues. This is a crucial capability for a variety of applications, such as emotional well-being evaluation and human-computer interaction (Table 1).

Table 1. Key Metrics and Configuration Details of the Proposed Real-Time Facial Emotion Recognition System

Metric	Data Size	Training Data	Testing Data	Epochs	Feature Detection	Optimizer	Model Accuracy	Real-time Detection
Value	19502	12436	7066	100	Haar Cascade	Adam	84%	Efficient

5 Results and Discussion

The outcomes of the proposed methodology demonstrate the effectiveness of the integrated approach in detecting facial emotions in real time. The CNN model, trained on a meticulously pre-processed dataset of 19,502 images, achieved an overall accuracy of 84% in classifying different facial expressions. This high accuracy highlights the robustness of our model in real-time applications, facilitated by the combined use of PyTorch and TensorFlow for smooth model creation and assessment. The Adam optimizer further accelerated the convergence process, enhancing the model's performance.

Compared to other state-of-the-art approaches, our method performs well in terms of accuracy and real-time processing. For instance, previous studies have achieved accuracy rates around 92.81% but often on static images or with limited real-time applicability. In contrast, our model not only maintains high accuracy but also ensures real-time responsiveness, a critical factor for practical applications.

Our approach uniquely combines the Haar-cascade algorithm for efficient face detection with a robust CNN architecture for emotion classification, ensuring real-time performance and high accuracy. This integration is novel compared to existing methodologies that either focus on static image analysis or lack the real-time processing capability. The results underscore the potential for future developments and applications in various domains, particularly in virtual reality, human-computer interaction, and mental health analysis.

5.1 Fear

The model's output for the Fear class shows how well it can identify facial emotions associated with fear. This is crucial for applications where recognizing fear is crucial, such as security or emotional health evaluations.

5.2 Neutral and Angry

The merging of both neutral and furious expressions into a single class demonstrates the model's ability to distinguish between the minute differences between these two emotional states. This is particularly valid in circumstances where it's imperative to recognize intense emotional states, like fury.

5.3 Sad

The model's predictions show how sensitive it is to facial expressions associated with sadness for the Sad class. Accurately identifying depression has benefits in a variety of fields, including computer-human empathetic communication and psychological tracking.

5.4 Happy

The Happy class outputs demonstrate the model's proficiency in identifying pleasant emotions. The ability to recognize happiness is a crucial aspect of emotion detection. Applications for this can be found in industries such as surveys of customer satisfaction and entertainment (Fig. 7).

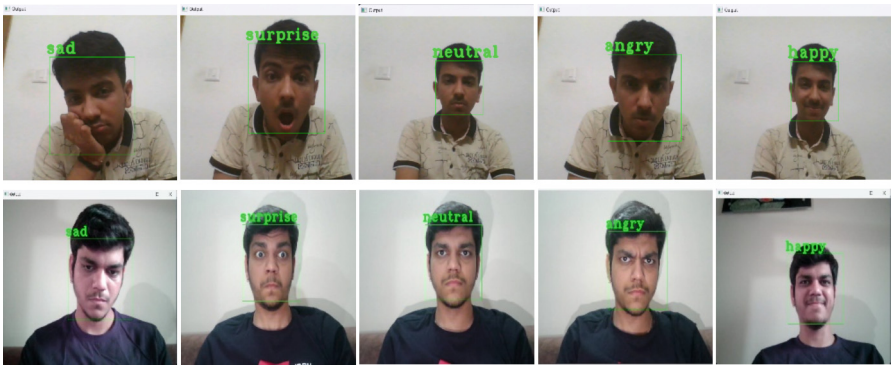


Fig. 7. Glimpse of some real-time model outputs

In Fig. 6, several frames from different testing scenarios are presented, showcasing the robustness and reliability of the model across diverse conditions and subjects. Each frame is annotated with the detected emotion and the corresponding confidence score, indicating the level of certainty in the model's predictions. The figure exemplifies the system's real-time processing capability, demonstrating minimal latency and high accuracy, which are critical for applications requiring immediate emotional feedback.

The visual evidence in Fig. 6 underscores the effectiveness of integrating the Haar-cascade algorithm for face detection with convolutional neural networks for emotion classification. The system's proficiency in handling variations in lighting, facial angles, and backgrounds is evident from the diverse set of images.

Table 2. Comparative Analysis of Method Performance in Real-Time Facial Emotion Recognition

Method	Accuracy (%)	Training Time	Real-time Detection
Method A	71	100 Epochs 226 Samples/Epoch	Slow
Method B	84	100 Epochs 98 Samples/Epoch	Efficient

In our study, we employed two distinct methods, Method A and Method B, to evaluate the performance of our real-time facial emotion recognition system. The comparison between these methods is summarized in Table 2. The reason behind not choosing Method A is because it was relatively slow due to the higher number of samples processed per epoch. The slower training process was a result of the larger batch size, which increased the computational load.

The comparative analysis of Methods A and B emphasizes how important it is to evaluate real-time face emotion recognition systems effectiveness in terms of detection speed, accuracy, and training configuration.

Accuracy vs. Training Configuration: Method A's higher accuracy allowed it to achieve performance that was either better than or equivalent to Method B's, even with Method B having a larger training sample size. This suggests that Method A's efficient model design and training procedure allowed for improved accuracy to be achieved with a relatively smaller training dataset.

Efficiency vs. Speed: In terms of instantaneous time detecting speed, Method B performs less efficiently than Method A. While both methods achieved the desired accuracy, Method A is a better option due to its responsiveness in scenarios where prompt emotion recognition is essential.

For Method B which is the proposed method, the model's accuracy was 84% with 100 training sessions (each containing 98 samples). The Adam optimizer allowed for the improvement of convergence efficiency and the minimizing of training loss. Real-time face identification using the Haar-cascade method showed how well it worked swiftly and accurately identifying different parts of the face. This integration substantially contributes to the system's real-time detection speed, which is an important feature for applications that need to recognize emotions quickly and accurately.

6 Conclusion

This study successfully integrated deep learning techniques with efficient face detection algorithms to develop a robust system for real-time facial emotion recognition. The comparative analysis demonstrated that the approach involving 100 epochs with 98 samples per epoch (Method B) significantly outperformed the alternative method (Method A) in both accuracy (84% vs. 71%) and efficiency. However, the system exhibits some weaknesses that should be addressed in future research. The model's performance was lower

in low-light conditions and with partially occluded faces, and the recognition of more subtle emotions, such as contempt, was less accurate. Future work should focus on incorporating additional modalities like audio and physiological signals to improve emotional classification, as well as optimizing the system for deployment on resource-constrained devices. By addressing these limitations and exploring new research avenues, the field of real-time facial emotion recognition can continue to evolve, offering increasingly accurate and versatile solutions for a wide range of applications.

7 Future Research

Even though the suggested real-time facial expression recognition system has shown encouraging results, there are a number of ways that future study can improve its capabilities. Including other modalities, like auditory and physiological signals, could offer a more thorough method of detecting emotions by utilizing a variety of data sources to enhance overall accuracy and more accurately identify subtle emotional indicators. Investigating how the deep learning architecture incorporates attention mechanisms may also aid in the model's ability to concentrate on the most important face features for precise emotion categorization, which may be especially helpful for identifying more complex emotional expressions.

The robustness and generalization of the model could be enhanced by extending the data augmentation approaches to include realistic occlusions and the generation of synthetic facial expressions. By optimizing the system for wear able and mobile platforms, emotion-aware applications can be used in a variety of scenarios and become more accessible and applicable in the real world. Further research on the ethical implications of real-time emotion recognition, including issues related to fair use and privacy, is crucial to ensure the responsible development and application of this technology.

References

1. Dalvi, C., Rathod, M., Patil, S., Gite, S., Kotecha, K.: A survey of AI-based facial emotion recognition: features, ML & DL techniques, age wise datasets and future directions. IEEE Access **9**, 165806–165840 (2021). https://doi.org/10.1109/ACCESS.2021.3131733
2. Dar, T., Javed, A., Bourouis, S., Hussein, H.S., Alshazly, H.: Efficient-SwishNet based system for facial emotion recognition. IEEE Access **10**, 71311–71328 (2022). https://doi.org/10.1109/ACCESS.2022.3188730
3. Heredia, J., et al.: Adaptive multimodal emotion detection architecture for social robots. IEEE Access **10**, 20727–20744 (2022). https://doi.org/10.1109/ACCESS.2022.3149214
4. Hwang, C.L., Deng, Y.C., Pu, S.E.: Human–robot collaboration using sequential-recurrent-convolution-network-based dynamic face emotion and wireless speech command recognitions. IEEE Access **11**, 37269–37282 (2023). https://doi.org/10.1109/ACCESS.2022.3228825
5. Kahou, S.E., et al.: EmoNets: multimodal deep learning approaches for emotion recognition in video (2015). https://arxiv.org/abs/1503.01800
6. Li, T.H.S., Kuo, P.H., Tsai, T.N., Luan, P.C.: CNN and LSTM based facial expression analysis model for a humanoid robot. IEEE Access **7**, 93998–94011 (2019). https://doi.org/10.1109/ACCESS.2019.2928364

7. Liliana, D.Y.: Emotion recognition from facial expression using deep convolutional neural network. J. Phys. Conf. Ser. **1193**(1), 012004 (2019). https://doi.org/10.1088/1742-6596/1193/1/012004
8. Pandey, A., Gupta, A., Shyam, R.: Facial emotion detection and recognition **7**, 176–179 (2022). https://doi.org/10.33564/IJEAST.2022.v07i01.027
9. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: British Machine Vision Conference (2015). https://api.semanticscholar.org/CorpusID:4637184
10. Sambare, M.: FER 2013 dataset. https://www.kaggle.com/datasets/msambare/fer2013. Accessed 28 Mar 2024
11. Shah, S., Tembhurne, J.: Object detection using convolutional neural networks and transformer-based models: a review. J. Electr. Syst. Inf. Technol. **10**(1), 54 (2023). https://doi.org/10.1186/s43067-023-00123-z
12. Shehada, D., Turky, A., Khan, W., Khan, B., Hussain, A.: A lightweight facial emotion recognition system using partial transfer learning for visually impaired people. IEEE Access **11**, 36961–36969 (2023). https://doi.org/10.1109/ACCESS.2023.3264268
13. Shetty, A.B., Bhoomika, Deeksha, Rebeiro, J., Ramyashree: Facial recognition using Haar cascade and LBP classifiers. Glob. Transit. Proc. **2**(2), 330–335 (2021). https://doi.org/10.1016/j.gltp.2021.08.044. https://www.sciencedirect.com/science/article/pii/S2666285X21000728. International Conference on Computing System and its Applications (ICCSA- 2021)
14. Veltmeijer, E.A., Gerritsen, C., Hindriks, K.V.: Automatic emotion recognition for groups: a review. IEEE Trans. Affect. Comput. **14**(1), 89–107 (2023). https://doi.org/10.1109/TAFFC.2021.3065726
15. Villegas-Ch, W.E., García-Ortiz, J., Sánchez-Viteri, S.: Identification of emotions from facial gestures in a teaching environment with the use of machine learning techniques. IEEE Access **11**, 38010–38022 (2023). https://doi.org/10.1109/ACCESS.2023.3267007
16. Vinh, T.Q., Anh, N.T.N.: Real-time face mask detector using YOLOv3 algorithm and Haar cascade classifier. In: 2020 International Conference on Advanced Computing and Applications (ACOMP), pp. 146–149 (2020). https://doi.org/10.1109/ACOMP50827.2020.00029
17. Yang, J., Qian, T., Zhang, F., Khan, S.U.: Real-time facial expression recognition based on edge computing. IEEE Access **9**, 76178–76190 (2021). https://doi.org/10.1109/ACCESS.2021.3082641
18. Zhang, H., Jolfaei, A., Alazab, M.: A face emotion recognition method using convolutional neural network and image edge computing. IEEE Access **7**, 159081–159089 (2019). https://doi.org/10.1109/ACCESS.2019.2949741
19. Zhang, X., Qi, G., Fu, X., Zhang, N.: Robust emotion recognition across diverse scenes: a deep neural network approach integrating contextual cues. IEEE Access **11**, 73959–73970 (2023). https://doi.org/10.1109/ACCESS.2023.3296316



Deepfake Detection Using ResNet50: Performance Analysis on Celeb-DF Dataset

Yashoda Alpesh Chouhan, Chetankumar Chudasama(✉), and Deepak Kumar Verma

Marwadi University, Rajkot, Gujarat, India
er.chetanchudasama@gmail.com

Abstract. As digital forgery blurs the line between truth and illusion, deep-fake technology presents unique problems for the integrity and reliability of online media. The use of fakes that can mimic real videos has sparked concerns about how they can be abused to spread false information and erode public confidence in real ones. As a result, it is important to ensure the authenticity of digital content by effectively detecting manipulated videos. In this study, we performed several experiments on ResNet50 trained using the Celeb-DF dataset using different hyperparameters and fine tuning methods to analyze them on the basis of metrics and evaluate its performance. Of the 12 various experiments performed, the model performed exceptionally well in some hyperparameters, achieving an AUC score of 98.11%. The findings of this article attempt to help build confidence in society in light of the growing threat posed by deepfakes.

Keywords: Celeb-DF · ResNet50 · Deepfake Detection · Deep Learning · Machine Learning

1 Introduction

Deepfake technology has lately gained interest as well as concern from the public. Advanced algorithms have enabled the manipulation of videos in such a way that influential individuals can be made to appear in places they have never gone to and say things that they have never said. The latest scenarios where deepfake has caused an increase in frauds were seen with the rise of digital doppelgangers. This technology can create identical copies of individuals in videos and can be used for payment fraud as well [1]. Social media platforms have witnessed situations whereby a deepfake video featuring a public figure spread showing them uttering controversial statements that incite public rage. The truth does not matter in such situations because fake videos can drastically manipulate public opinion and cause immense damage. Similarly, manipulated videos showcasing image-tarnishing actions such as unethical or compromising activities of people in the form of memes are harmful to the individuals involved. Such incidents show the serious dangers posed by fake technology in society. Scalable methods are needed to find Deepfakes on a large scale, and deep learning models are perfect for this. However, these models need training data, and while it is not hard to make a few Deepfakes that look real, the cost of making the hundreds of thousands of videos that

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 397–404, 2025.
https://doi.org/10.1007/978-3-031-86296-0_28

these models need to be trained is often too high [3]. Therefore, in this study we use the Celeb-DF dataset [4]. This wide data set helps us to train ResNet50 effectively and deliver the desired results. The study aims to showcase methods for deepfake detection with the help of deep learning models. The goal of this paper is to develop an algorithm for deepfake detection with the help of deep learning.

The first author implemented the project, performed the relevant research, and composed the article. Throughout the process, the second author gave helpful advice and guidance. The third author provided support and motivation that helped the project as a whole.

This research is organized as follows: Sect. 2 presents the literature review of previous work done, Sect. 3 discusses the proposed method. Results and discussions are in Sect. 4 and the paper concludes in Sect. 5.

2 Related Works

As technology behind deepfakes has advanced, research on deepfake detection has made significant strides. In the beginning, standard machine learning methods were used mainly for the identification of visual defects. An example is presented by the work of Korshunov and Marcel, which used facial artifacts as well as inconsistencies to detect manipulated videos [5]. Detection using CNNs became popular with the development of better deep-fake techniques. Among them was Chollet's XceptionNet that applied deeper CNN architectures to improve detection accuracy by capturing more intricate visual details [6]. The release of the Celeb-DF dataset by Yuezun et al. has also moved deeper into this area by providing a high resolution reference for the evaluation of detection methods to provide stronger performance assessments [4]. A remarkable work was done by Shwetambari Borade et al., they trained the Celeb-DF dataset on ResNet50 alongside image scraping technology which resulted in marvelous results [7]. Kaiming et al. introduced a groundbreaking neural network design called ResNet that trains very deep networks in an easy way by using shortcut connections which we will use further in this study [8]. Intensive research on deepfake detection highlights the need for constant updates and innovations due to the rapidly growing advances in the creation of deepfakes.

3 Proposed Method

In this research work, we utilize ResNet50 architecture for detecting deepfakes trained on the Celeb-DF dataset by upgrading a pre-trained model based on our specific needs and fine-tuning them using the deep residual learning framework of Resnet50. Therefore, the model consists of a global average pooling layer, a densely connected layer with a range of neurons, and a ReLU activation function, a dropout layer to avoid overfitting effects, ultimately resulting in a sigmoid output layer for binary classification. To optimize performance during training, hyperparameters such as epochs (50, 80), batch size (32, 64), and dense layer configuration (128, 256, 512) are altered through the 12 experiments done. Evaluation is performed on accuracy metrics including precision, recall, F1 score, ROC AUC score, as well as confusion matrices and graphs to visualize and better understand the results and results.

3.1 Dataset

There are several datasets available for deep-fake detection. However, what makes Celeb-DF a better choice is that the real videos have many different modifications, such as the faces of the people in them being different sizes (in pixels), positioned differently, lighted differently, and the backgrounds being different which is useful in efficiently training the model for high-yield outputs. There are 590 real videos and 5,639 DeepFake videos in the Celeb-DF collection, which adds up to more than two million video frames [4]. The classification of the dataset is shown in Fig. 1. From this dataset we have used a total of 1200 videos of which 12000 frames were extracted (6000 fake and 6000 real).



Fig. 1. Snippet of the Celeb-DF dataset divided by class [9]

3.2 ResNet50

ResNet50 is a deep convolutional neural network known for its way of training deep models as presented in 2015 by Kaiming He et al. [8] It consists of residual blocks with shortcut connections by which the network hops over certain layers. Through this design, it addresses standard issues relating to training such as vanishing gradients, hence allowing it to learn efficiently on intricate patterns. The model comprises 50 layers and incorporates bottleneck layers that help create a balance between depth and computational efficiency which is shown in Fig. 2. For the purposes of accuracy and effectiveness of feature extraction tasks, ResNet50 remains manageable [8].

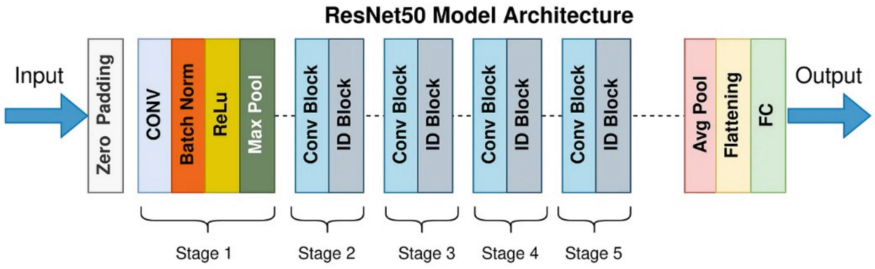


Fig. 2. Basic CNN architecture of ResNet50 model [2]

ResNet50's competence is key to deepfake detection. This is because it can capture fine details and aspects in images, which are critical in distinguishing real and manipulated content. ResNet50 is among the best algorithms for deepfake detection, as it accurately identifies real versus fake images.

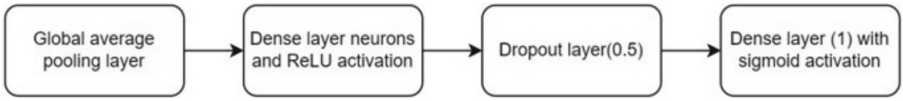


Fig. 3. Additional layers added in the Transfer Learning process

The model was passed through additional layers such as the global average pooling layer followed by the dense layer with ReLU activation, dropout layer, and concluded with a dense layer with sigmoid activation of the transfer learning process, as shown in Fig. 3.

3.3 Data Preprocessing

The Celeb-DF data set was preprocessed by converting the videos into frames using the extraction method. This step helped to make sure the data were in a manageable format to train and handle. Each video was processed to extract 10 face frames to ensure that sufficient data was collected. After extracting these frames, the data were divided into three folders: train, test and validation with the 6: 3: 1 ratio, respectively, as shown in Table 1. The files were then loaded into the model using data loaders to speed up batch processing and employ data augmentation methods, which improved the performance of the model.

Table 1. Dataset division(frames)

	Train	Test	Val
Fake	3600	1800	600
Real	3600	1800	600

This structured process made sure that the model could learn from and evaluate a wide range of data, which made it possible to find deepfakes accurately.

3.4 Experiment

The purpose of the experiment was to train ResNet50 using the Celeb-DF dataset under different combinations of hyperparameters and analyze which combination gives the best outcome. In the first three experiments, the epochs and batch size were set to 50 and 32 respectively, while the neurons dense layer were 128, 256 and 512 accordingly. The AUC scores for these three experiments ranged between 96.87% and 97.23%. The

hyperparameters of the following three experiments were set as 50 epochs and 64 batch size, while the neurons dense layer were 128, 256 and 512 respectively. The AUC scores for these three experiments ranged between 96.77% and 96.99%. For the further three experiments the epochs and batch size were set to 80 and 32 respectively while the neurons were 128, 256 and 512 accordingly. The AUC scores for these three experiments ranged between 96.77% and 98.11%. Lastly, the set combinations for the three experiments remaining were set to 80 epochs, 64 batch size, while neurons were 128, 256 and 512 respectively. The AUC scores for these three experiments ranged between 96.61% and 96.68%.

4 Results and Discussions

Out of all the 12 experiments carried out on the Celeb-DF dataset using the ResNet50 architecture. The results obtained were exceptional, with the highest AUC of 98.11%. Conversely, the minimum AUC obtained was 96.61% attesting to a good performance but highlighting variances introduced as a result of diverse hyperparameter settings.

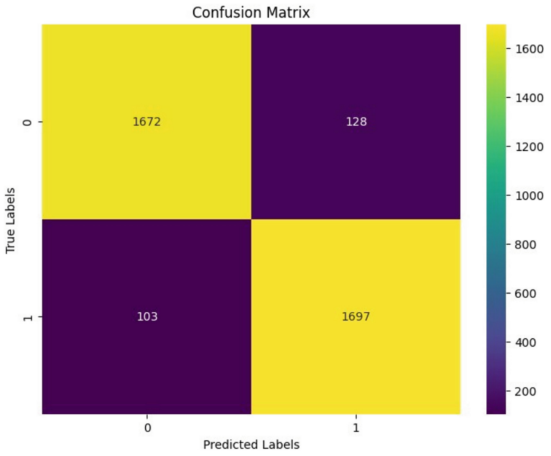


Fig. 4. Confusion Matrix (Experiment that achieved highest AUC score)

The confusion matrix of the highest AUC and the training and validation graphs for accuracy and loss achieved are shown in Fig. 4 and Fig. 5 respectively.

4.1 Comparitive Analysis Using Different Hyperparameters

We trained the ResNet50 model 12 times in deepfake detection using the Celeb-DF dataset with different hyperparameter combinations to explore how different settings would affect its performance. From these experiments, we were able to attain a exceptional AUC score of 98.11%. Such findings reveal that hyperparameter tuning significantly improves model's efficacy. We were able to drastically improve the detection

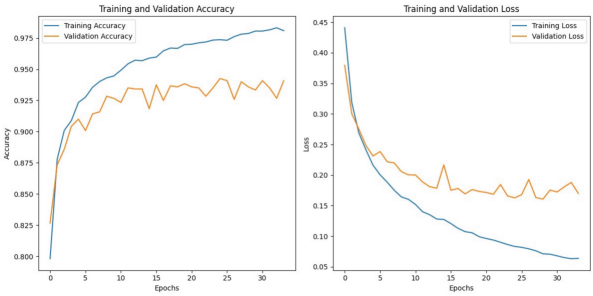


Fig. 5. Training and validation graphs for accuracy and loss (Experiment that achieved highest AUC score)

capabilities of ResNet50 by changing learning rates, batch sizes and network architectures among others. The best result was obtained when we set the hyperparameters set at 80 epochs, 32 batch size and 128 neuron dense layer. The same hyperparameter setting achieved the highest Accuracy, Recall and F1 Score 93.58%, 94.28% and 93.63% respectively. However the combinations that resulted in the highest precision of 95.49% was when the settings were 80 epochs, 32 batch size and 512 neuron dense layer. Table 2 shows all the research findings.

Table 2. Analysis of ResNet50 on performance metrics by % (Trained under different combinations of hyperparameters)

Epochs	Batch size	Neurons ROC	AUC Score	Accuracy	Precision	Recall	F1 Score
50	32	128	96.87	91.18	87.10	92.96	89.94
50	32	256	97.23	92.08	88.00	94.18	90.98
50	32	512	97.05	91.56	88.30	92.35	90.28
50	64	128	96.99	91.56	88.08	92.66	90.31
50	64	256	96.99	90.92	87.24	92.04	89.58
50	64	512	96.77	90.53	85.87	92.96	89.28
80	32	128	**98.11**	**93.58**	92.99	**94.28**	**93.63**
80	32	256	97.35	92.08	88.21	93.88	90.96
80	32	512	96.77	91.94	**95.49**	92.16	93.80
80	64	128	96.61	90.82	95.21	90.67	92.89
80	64	256	96.68	91.24	95.34	91.20	93.23
80	64	512	96.65	91.24	95.14	91.41	93.24

4.2 Comparative Analysis with Other Researcher Findings

The highest AUC Score achieved in the experiments carried out on ResNet50 model was then compared further to other research findings that had used the Celeb-DF detaset for deepfake detection as shown in Table 2. The specific technique we used in our research lead to an outstanding AUC score of 98.11% as compared to other studies. For instance, Arini et al. utilized a hybrid of Xception and ResNet50 models together with Local Binary Pattern (LBP) characteristics, which perhaps constrained their performance in comparison to our enhanced ResNet50 model [10]. Rössler et al. used multiple models for facial manipulation but did not completely use ResNet50's advanced capabilities [11]. Radford et al. focused on transferable visual models, not specifically tuned for deepfake detection [12]. Yan et al. (n.d.) aimed at feature generalization, which might have compromised task-specific accuracy [13]. Our focused ResNet50 fine-tuning led to higher deepfake detection in terms AUC Score (Table 3).

Table 3. Results compared with other findings

Reference	AUC Score
In study [10]	0.87
In study [10]	0.81
In study [11]	0.96
In study [12]	0.96
In study [13]	0.97
Ours	0.98

5 Conclusions

The continuous fast-paced advancements in the creation of deepfakes and the simplicity with which deepfakes can be created with no expertise required leads to a plethora of threats that come along with it. It requires the deepfake detection technology to emerge with the same pace as well in order to restore trust in the society and avoid malpractices that happen on a large scale due to unethical use of deepfakes. Our study commenced with training ResNet50 on the Celeb-DF dataset under various set conditions to achieve the best results that are effective in detecting deepfakes accurately. The results from this study suggest that it is possible for deep learning models to be fine-tuned so as to have high metrics values in hard tasks like detecting deepfakes. In addition, future studies should take into account alternative models and advanced tuning techniques in order to make deepfake detection systems more reliable and efficient over time. Different datasets and models along with combinations of hyperparameters can be experimented with to achieve the desirable outcomes.

Disclosure of Interests. The authors have no competing interests to declare that they are relevant to the content of this article.

References

1. Upcoming Trends in Deepfake Technology, 5 April 2024. https://www.knowledgenile.com/blogs/future-trends-in-deepfake-technology-what-to-expect
2. https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758
3. Dolhansky, B., et al.: The DeepFake detection challenge (DFDC) dataset. https://arxiv.org/pdf/2006.07397 (2020)
4. Li, Y., Yang, X., Sun, P., Qi, H., Siwei: Celeb-DF: a large-scale challenging dataset for DeepFake forensics. In: IEEE Conference on Computer Vision and Patten Recognition (CVPR) (2020)
5. Korshunov, P., Marcel, S.: DeepFakes: a new threat to face recognition? Assessment and detection. https://arxiv.org/pdf/1812.08685v1.pdf (2018)
6. Chollet, F.: Xception: deep learning with depthwise separable convolutions (2017). https://doi.org/10.1109/cvpr.2017.195
7. Borade, S., et al.: ResNet50 DeepFake detector: unmasking reality. Indian J. Sci. Technol. **17**(13), 1263–1271 (2024). https://doi.org/10.17485/ijst/v17i13.285
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2016). https://doi.org/10.1109/cvpr.2016.90
9. Lacerda, G.C., Da Silva Vasconcelos, R.C.: A machine learning approach for DeepFake detection. https://arxiv.org/abs/2209.13792, 28 September 2022
10. Arini, A., Bahaweres, R.B., Haq, J.A.: Quick classification of Xception and resnet-50 models on deepfake video using local binary pattern (2022). https://doi.org/10.1109/ismode53584.2022.9742852
11. Rössler, A., Cozzolino, D., Verdoliva, L., Thies, J., Nießner, M.: FaceForensics++: learning to detect manipulated facial images (2019b)
12. Learning Transferable Visual Models From Natural Language Supervision (n.d.). https://proceedings.mlr.press/v139/radford21a/radford21a.pdf
13. Yan, Z., Zhang, Y., Fan, Y., Wu, B.: UCF: uncovering common features for generalizable deepfake detection. In: The School of Data Science, the Chinese University of Hong Kong, Shenzhen (CUHK-Shenzhen), China (n.d.)



Performance of Linear and Polynomial Kernels of SVM Towards Kidney Cyst Detection

Rahul Nimbai, Mayur Patil, Veereshkumar Rathod, Santosh Pattar, and Prema T. Akkasaligar(✉)

Department of Computer Science and Engineering,
KLE Technological University's Dr. MSSCET, Belagavi, Karnataka, India
premaakkasaligar.mss@kletech.ac.in

Abstract. This research looks into the application of Support Vector Machines (SVMs) for image classification within the medical field, focusing on kidney cyst detection in CT-scanned images. It investigates the effectiveness of SVMs with different kernels in distinguishing between images containing cysts and those depicting normal kidneys. This research seeks to develop and evaluate the performance of SVM models using linear and polynomial kernels to classify kidney cysts in CT scan images. However, the presence of non-linear patterns in the data can pose challenges for traditional machine learning algorithms. The models carried out are trained on a dataset of CT-scanned images labeled as cysts or normal. A comparison using performance metrics, including accuracy, precision, recall, and F1 score is done to determine the most effective model for cyst classification. The linear kernel SVM achieved an accuracy of 86.90%, demonstrating good performance in classifying both cyst and normal images, whereas the polynomial kernel SVM outperformed the linear model, achieving a higher accuracy of 89.74%. Our findings suggest that the polynomial kernel SVM outperforms the linear kernel SVM for classifying kidney cysts in CT-scanned images. The polynomial kernel effectively captures non-linear patterns, leading to improved accuracy and maintaining efficiency. The model can help a professional in the diagnosis of an individual by keeping its own classification in front.

Keywords: Computer-aided Diagnosis · Hyperparameter Tuning · Kidney Cysts · Performance Analysis · SVM Kernels

1 Introduction

Kidney cysts, which are fluid-filled sacs that develop in the kidneys, are typically symptom-free. They originate from the abnormal expansion of tiny kidney tubes. While most are simple cysts, genetic conditions (like polycystic kidney disease can cause numerous cysts and complications like high blood pressure, flank pain, and blood in the urine). If a cyst breaks, bleeds or grows large, symptoms like fever, frequent urination, or abdominal pain appear. Though the exact cause of simple cysts is unknown, age, injury, and even blockages in kidney tubes contribute towards their formation [1]. Kidney cysts, particularly simple cysts, do not have stages like cancer. Simple cysts are benign and

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 405–416, 2025.
https://doi.org/10.1007/978-3-031-86296-0_29

do not progress in a predictable way that would warrant staging. While complex cysts may require further evaluation due to irregular features, they also do not follow a formal staging system. Therefore, classifying cysts into stages is not applicable. The goal of this research is to detect the presence of cysts, which aligns with standard clinical practices for kidney cyst evaluation.

Kidney Computed Tomographic (CT) images [2] provide valuable information about kidney health, however checking for the presence of any disease, manually has constraints like time and quality. Automated image analysis using machine learning offers a promising objective for efficient analysis and diagnosis. This research focuses on developing and comparing machine learning models for classifying kidney CT images containing cysts from normal healthy kidneys [3]. Reason for choosing CT images over any other images is that they are state-of-the-art X-rays that use emissions to capture detailed images of bones, blood vessels, and even air pockets, making them great for examining complex structures and spotting fractures, tumors, or cysts.

This work proposes the use of SVM featuring diverse kernels like linear and polynomial, to classify kidney cysts in CT scans SVM is a versatile supervised learning technique widely used for tasks such as classification and regression. It excels in high-dimensional environments and tends to avoid overfitting, especially when there are more dimensions than samples. Compared to deep learning methods, SVMs offer a clearer interpretation while still achieving impressive in various classification challenges. The goal here is to assess how these SVM models perform with different kernels and determine their effectiveness in identifying cysts. A detailed comparison will analyze these SVM models based on accuracy, precision, recall, plus other important metrics.

Convolutional Neural Networks (CNN), a Deep Learning (DL) technique, have emerged as the leading approach for medical image analysis, achieving high accuracy in cyst detection. However, their complex nature often leads to a lack of interpretability, hindering their clinical adoption due to concerns about understanding the decision-making process behind the results. The CNNs offer impressive performance, but their "Black Box" nature raises concerns about their interpretability and reliability in clinical settings. Additionally, training a DL model often requires large amount of data and computational resources, which might not be readily available in all healthcare settings. Accurately identifying kidney cysts in CT scans presents several challenges. The diverse shapes, sizes, and intensities of cysts themselves create ambiguity. Additionally, noise and artifacts introduced during image acquisition further complicate analysis. Finally, limited access to large, well-labeled datasets in some healthcare settings hinders the development and implementation of robust image analysis tools.

In this research, the application of SVM models with both linear and polynomial kernels for the detection of kidney cysts in CT scanned images is explored. While SVMs have been widely used for various classification tasks, our study uniquely contrasts the performance of linear and polynomial kernels to determine the most effective approach for this medical image processing application. The uniqueness of our research is rooted in the analysis and comparison of these kernels, with our results showing that the polynomial kernel significantly surpasses the linear kernel in precision, recall, and accuracy. Future research could explore several directions to enhance kidney cyst detection using Support Vector Machines (SVM). One area could incorporate additional kernel function

types, such as Radial Basis Function (RBF) or sigmoid kernels, to improve non-linear data classification. Exploring hybrid models that integrate deep learning approaches with SVM may also provide better accuracy and efficiency, particularly in medical image analysis. Lastly, expanding the dataset to include images from different populations and using cross-validation strategies would help enhance the model's generalizability in real-world clinical settings.

The rest of the paper is structured as follows: Sect. 2 reviews the existing work on the classification of kidney cysts. In Sect. 3 we outline our problem statement and objectives, providing a detailed description of our SVM model and the various kernels employed for classification. Section 4 presents the results and offers a discussion of the findings. Finally, Sect. 5 provides concluding remarks.

2 Literature Survey

The emergence of Machine Learning (ML) and DL models offers a promising avenue for automating cyst detection and analysis, it leads to early diagnosis, improved accuracy, and enhanced workflow efficiency.

Ferjani et al. [4] proposed disease prediction using machine learning by focusing on its application in detecting kidney diseases and Parkinson's disease. The methodology employed includes utilizing SVM, kNN, and decision trees as a classification algorithm to identify these specific diseases. The SVM model includes its reliability in handling high-dimensional, semi-structured, and unstructured data, making it superior in accuracy for kidney diseases and Parkinson's disease detection. The accuracy of the SVM model for kidney diseases and Parkinson's disease detection is 95.85% emphasized as being notably high in the study. Specifically, the SVM algorithm achieved higher accuracy compared to other classification algorithms used in the research. Shailja Rana et al. [5] proposed a computer-aided diagnostic (CAD) system for classifying focal kidney lesions, specifically differentiating between benign angiomyolipomas (AML) and malignant renal cell carcinomas (RCC), using wavelet-based texture descriptors. The methodology involves analyzing a dataset of 47 ultrasound images (22 AML and 25 RCC), where areas of interest (AOIs) are manually extracted and various wavelet filters, including Daubechies and Symlets, are applied to derive texture features. The classification is performed using a Support Vector Machine (SVM) classifier, achieving an overall accuracy of 82.6%, with individual accuracies of 63.6% for AML and 100% for RCC.

The survey of research papers reveals the significant potential of DL models for kidney cyst detection, achieving promising accuracy and efficiency comparable to traditional machine learning models, and in some cases exceeding, human experts. However, several challenges remain, including the need for larger, diverse datasets, improved generalizability across imaging modalities, and the integration of these models into clinical workflows. Table 1 summarizes the above works and compares them. In this work, a lightweight SVM kernel for the classification of kidney cysts to overcome the above challenges is developed.

Table 1. Summary of literature review

Reference	Approach	Dataset	Performance
Mardison et al. [6]	Hybrid model for cyst detection using ANN	Hospital patients, 25 Ultrasound images	Accuracy: 84%
Qin et al. [7]	Computer-aided measurements of simple kidney cyst	UCI repository	Accuracy: 99.83% Sensitivity: 99.84% Specificity: 99.80% F1-score: 99.86%
Kline et al. [8]	Automated semantic segmentation	60 patients with ADPKD	Mean score: 85%
Chen et al. [9]	AHDCNN	The Dataset have been taken from [10]	Accuracy: 97.30% F1-score: 97.30%
Sudharson et al. [11]	SVM classifier with SVD features	Images are taken from the ultrasound database [20, 21]	Accuracy: 89% Recall: 90%
Hossain et al. [12]	Proposed an SVM model for classification	The dataset comprises 12,427 images	Accuracy LBP: 89% Accuracy ALBP: 99%
Chowdhury et al. [13]	Mutual learning model using the CNN and SVM	The dataset consists of 12,446 kidney images	Accuracy of SVM, Linear: 76.59% Non-linear: 85.90%

3 Problem Statement and System Model

In this section, the problem statement for the detection of kidney cysts in CT images and its objectives along with the description of our ML-based model for kidney cyst detection using different types of SVM kernels are presented.

3.1 Problem Statement

A kidney cyst is a fluid-filled sac that forms on or within the kidneys. These cysts can vary in size, from small and barely detectable to quite large, and may appear as a single cyst or in multiple clusters. Most kidney cysts are simple cysts, that are harmless and typically don't cause any symptoms, so designing and training a machine-learning model optimized for kidney cyst detection, using a variety of SVM kernels is our aim in this work.

3.2 Objectives

Objectives of the proposed work are as follows:

- To design and train an SVM model that achieves high accuracy for classifying images as containing simple kidney cysts or not.
- To explore the effectiveness of different SVM kernels for this specific task of kidney cyst detection.

3.3 SVM

SVM or Support Vector Machine, is a type of supervised learning algorithm's useful for both classification and regression tasks. Essentially it identifies a hyperplane (a linear boundary) that separates various classes. This separation happens with the maximum margin possible in the feature space. The significance of this margin is confidence. More confidence often leads to better generalization when dealing with unseen data.

In SVM, classification occurs through a two-step process in a high-dimensional feature space. Figure 1 depicts our workflow of the proposed model for kidney cyst detection using the above-described SVM principles, the algorithm 1 discusses the entire steps involved in the process of kidney cysts detection. Firstly, feature extraction transforms the input data into points within this space. Next, the SVM methodology aims to find the optimal hyperplane, which serves as a decision boundary in the feature space, maximizing the margin between different classes. The points nearest to this boundary, known as support vectors, are crucial in determining the hyperplane's position. New data points are then classified based on their position relative to the hyperplane: if they fall on one side, they are assigned to one class; if on the other, they are assigned to the opposite class. This method makes SVMs particularly robust against outliers and effective at handling complex data, even when the initial feature space is not easily separable.

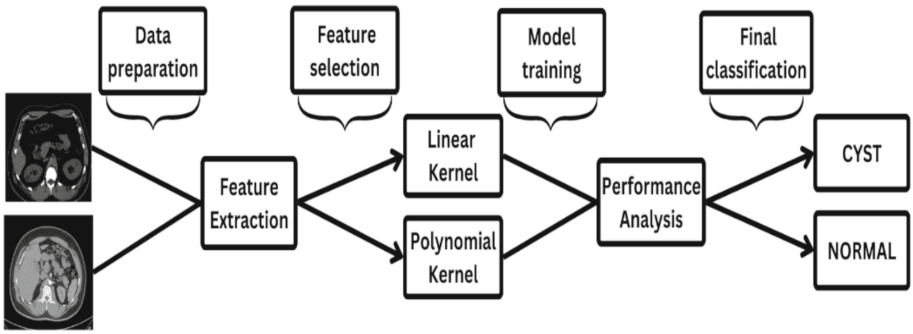


Fig. 1. Workflow diagram of Proposed Model

Algorithm 1 Classification of Kidney Cysts

Input: Reading CT images of Kidney Cysts
Output: Classification of cysts/ normal image identification

Start
Step 1: Preprocessing / Resizing
Step 2: Data augmentation to perform balancing of dataset
Step 3: Loading augmented images
Step 4: Perform train test split as 80:20
Step 5: Perform feature engineering by converting 3D array to 1D array
Step 6: If (option == 1) then Perform training using SVM with linear kernel Else Perform training using SVM with polynomial kernel endif
Step 7: Evaluate the model using training time and test time by varying the lasso regularization hyperparameter C from C = 10 to C = 10000
Step 8: Calculate the model performance using metrics like accuracy, precision, recall, f1-score, and AUC
Step 9: Classify image as kidney cyst or normal kidney
End

Parameters that are taken into consideration by SVM:

- Kernel: SVMs can manage non-linear data using kernel functions that map data points to the higher dimensions where a linear separation becomes possible. Common kernels include linear, polynomial, and radial basis function (RBF).
- C (Regularization Parameter): This parameter controls the balance between maximizing the margin and permitting some misclassifications to achieve a better overall fit. Higher C means a stricter margin and lower tolerance for errors, while lower C allows for smoother decision boundaries but might be more prone to overfitting.

Potential advantages of choosing SVM:

- High-Dimensional Feature Space: The dataset of medical images comprises a high-dimensional feature space. SVM is well-regarded for its effectiveness in handling high-dimensional data, making it a suitable choice for our analysis.
- Robustness to Overfitting: The regularization parameter C in SVM plays a key role in controlling overfitting. By adjusting C, one can find the right balance between maximizing the margin and minimizing classification errors, which is essential for improving the model's generalization to unseen data.
- Memory Efficiency: SVM provides memory efficiency using a subset of training points in the decision function (support vectors).

3.4 Kernels Used

Linear Kernel. In the realm of SVM, the linear kernel stands out as the simplest and most interpretable option. The data points are inherently linearly separable in the original feature space, making it ideal for kidney cyst diagnosis. The linear kernel for kidney image of size (x_1, x_2) is:

$$K(x_1, x_2) = x_1^T * x_2 \tag{1}$$

where, $x_1^T * x_2$ denotes the dot product between x_1 and x_2. The linear kernel makes use of the feature space where the data points reside. The support vectors, which are the data points closest to the hyperplane, define the margin. The regularization parameter C influences the offset of the hyperplane from the origin, affecting how the model balances margin maximization and classification accuracy.

Polynomial Kernel. The Polynomial kernel in SVM addresses non-linear data by implicitly mapping it to a higher-dimensional space, making it possible to achieve linear separation within that transformed space. This "magic" happens through the kernel trick, allowing the model to operate in the original space without explicitly computing the mapping. The polynomial kernel for kidney image of size (x_1, x_2) is:

$$K(x_1, x_2) = (\gamma * x_1^T * x_2 + C)^d \tag{2}$$

The polynomial kernel also makes use of feature space, kernel space which is a higher-dimensional space created while performing operations, kernel trick, a technique that allows kernel functions to operate in the original feature space without explicit mapping. Also, degree (d) a parameter of the polynomial kernel controlling the complexity of the decision boundary, and lastly gamma (γ) and bias (C) are additional parameters influencing similarity calculations and classification margins are analyzed.

4 Results and Discussions

In this section, the implementation of the proposed workflow for kidney cyst detection using linear and polynomial kernels is discussed. Further, the experimental setup and results obtained are described.

4.1 Experimental Setup

The experiments are carried out on a machine equipped with an Intel Core i5 processor, and 8GB RAM NVIDIA GeForce GTX graphics card. The machine learning model development and evaluation include NumPy for efficient numerical computations and array manipulation. Scikit-learn is used to implement the machine learning algorithms SVM and provide evaluation metrics such as accuracy and F1-score. Matplotlib and Seaborn for creating visualizations to analyze model performance and data distributions. The dataset [14] is collected from the Picture Archiving and Communication System (PACS) at different hospitals in Dhaka, Bangladesh, where patients are already diagnosed with kidney conditions using Computed Tomography (CT) scans. The dataset is divided into two classes, where the normal kidney class contains 5,077 images and the kidney with cyst class has 3,709 images. The images are intended for use in medical image analysis and computer vision applications, particularly for training and evaluating machine learning models for kidney condition classification, The sample images of normal and cystic kidneys are shown in Fig. 2 and Fig. 3 respectively.

The dataset shows a class imbalance in the number of normal kidney images when compared to cyst kidney images. To overcome class imbalance data augmentation technique is used to balance and utilize both classes efficiently. The dataset is divided into 80% for model training and 20% for testing.

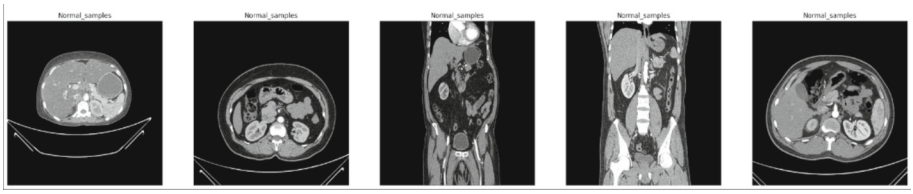


Fig. 2. Sample Kidney Normal Images



Fig. 3. Sample Kidney Cyst Images

4.2 Performance of the Linear Kernel

A linear SVM model takes a unique approach to learning from an image dataset. It focuses on finding the best hyperplane. This hyperplane separates the two classes within the dataset of CT images. One class shows kidneys with cysts, while the other class represents normal kidneys without cysts. In image classification, there are usually high-dimensional feature vectors for each image. Each feature indicates a pixel's intensity or describes something extracted from the image itself. The linear SVM algorithm works by classifying these feature vectors. It identifies the hyperplane that maximizes the margin. In simpler terms, this margin is the distance between that hyperplane and the closest support vectors from each class.

Table 2 shows that learning and testing time decreased as the value of lasso regularization parameter (C) increases because a higher value of C corresponds to a lower regularization strength, potentially leading to a simpler decision boundary. This classified 690 out of 869 cyst images (true positives) and 836 out of 1015 normal kidney images (true negatives) correctly. The Fig. 4 indicates misclassified 179 cyst images as normal kidney images (false negatives) and 51 normal kidney images as cysts (false positives) whose effects can be observed in Table 3 that the precision is 94.25% which is less when compared to the polynomial kernel.

Table 3 highlights the model's performance, with an accuracy of 86.90%, and summarizes the performance of both kernels for disease diagnosis. It is observed that the polynomial kernel outperforms the linear kernel concerning all the performance metrics. The Precision-Recall (PR) curve and Receiver Operating Characteristic (ROC) curve in Fig. 5, which according to the values obtained from the confusion matrix in Fig. 4, it is observed that the Area Under Curve (AUC) is lower, which indicates the low performance of the linear model.

Table 2. Learning and testing time of linear and polynomial kernel

C	Linear Kernel		Polynomial Kernel	
	Learning Time (in sec)	Testing Time (in sec)	Learning Time (in sec)	Testing Time (in sec)
10	137.19	115.87	145.435	126.132
100	128.41	115.43	147.241	129.004
1000	141.01	107.79	148.353	128.800
10000	122.75	108.10	145.580	131.751

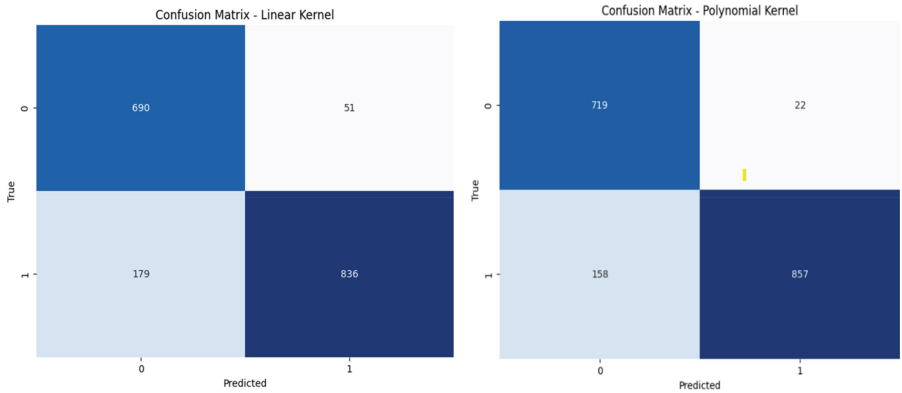


Fig. 4. Confusion Matrix of Linear and Polynomial Kernel

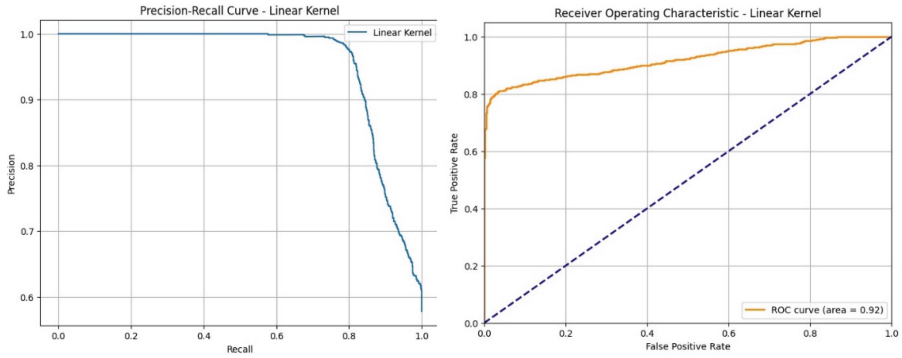


Fig. 5. Precision - Recall Curve and Receiver Operating Characteristics of Linear Kernel

4.3 Performance of Polynomial Kernel

A polynomial SVM model learns from an image dataset in a way that's somewhat like a linear SVM. However, it has this extra ability to understand non-linear relationships

Table 3. Performance of linear and polynomial kernel

	Accuracy	Precision	Recall	F1 Score
Linear Kernel	86.90	94.25	82.36	87.90
Polynomial Kernel	89.74	97.49	84.43	90.49

between different features. In medical image classification, each image turns into a high-dimensional feature vector. The polynomial SVM aims to identify a decision boundary that can adaptively separate classes. This is done by fitting a polynomial function to the dataset. It transforms the input features into a higher-dimensional space for this purpose. Here, the polynomial kernel calculates the inner products between feature vectors, allowing for more complex relationships to be captured effectively.

The average accuracy achieved is 89.74% which is shown in Table 3. However, accuracy remained almost constant for all C values between 10 and 10000. Precision remained very high (about 97.49%). Recall also stayed consistent (around 84.43%). F1 score followed a similar pattern, with 90.32%. Table 2 shows a learning time increase from 145.435 s for C = 10 to 148.353 s for C = 10000. The change in testing time is minimal across different C values, ranging from 126.132 s for C = 10 to 131.751 s for C = 10000. The Fig. 4 indicates the reduction in misclassifications that are caused when using the polynomial kernel. This suggests that the complexity of the decision boundary has little impact on prediction time.

When it comes to the PR curve and ROC curve of the polynomial kernel which are shown in Fig. 6, here it is observed that the AUC is greater which indeed indicates that the model can properly differentiate between the positive and negative classes and will have better and trustworthy predictions.

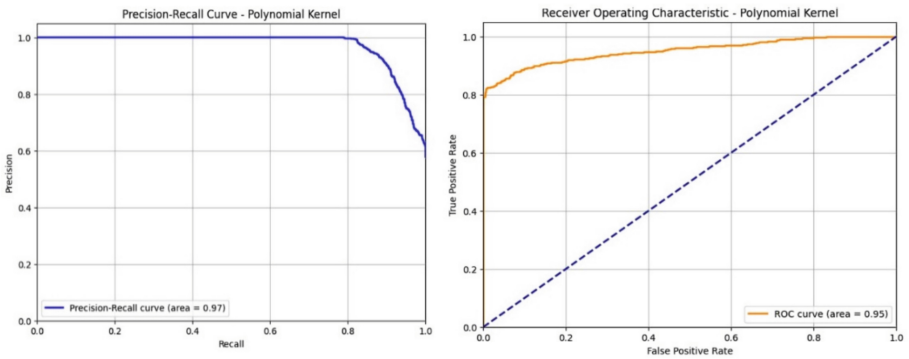


Fig. 6. Precision - Recall Curve and Receiver Operating Characteristics of Polynomial Kernel

Table 4 compares the performance of the proposed model with the other state-of-the-art SVM models of S Sudharson et al. [10], T Hossain et al. [11] and S Chowdhary et al. [12]. The proposed model is observed to have better accuracy than the other models.

Table 4. Comparison of the proposed model with state-of-the-art works

	Accuracy	Precision	Recall	F1 Score
S Sudharson et al. [10]	89	–	90	–
T Hossain et al. [11]	89	93	96	85
S Chowdhary et al. [12]	85.90	–	–	–
Proposed Model	**89.74**	**97.49**	84.43	**90.49**

5 Conclusions

In this study, evaluation of the performance of SVM models is done with linear and polynomial kernels for classifying CT images of kidneys as either containing cysts or being normal. The linear kernel SVM achieved an accuracy of 86.90%. This demonstrates notable success in correctly classifying both cyst and normal kidney images, with a relatively higher rate of false negatives. Conversely, the polynomial kernel SVM outperforms the linear model, achieving a higher accuracy of 89.74%. Precision remained consistently high across various C values, indicating a low rate of false positives, while recall and F1 scores also maintained robust levels. The polynomial SVM showcases superior performance in capturing non-linear patterns present in the data, resulting in more accurate classifications. Furthermore, the minimal increase in testing time across different C values suggests that the complexity of the decision boundary has less impact on prediction time, highlighting the efficiency of the polynomial SVM approach. These results are promising for developing a lightweight kidney cyst detection tool.

References

1. Leslie, S.W., Garfield, K.: Simple renal cyst. StatPearls [Internet]. Treasure Island (FL). StatPearls Publishing, January 2024 (2023). https://www.ncbi.nlm.nih.gov/books/NBK499900/
2. Akkasaligar, P.T., Pattar, S., Uppin, A., Kori, K., Kulakarni, A.: Kidney tumor detection using computed tomography scan images through CNN. In: 2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT). IEEE (2024)
3. Tsugaya, M., Kajita, A., Hayashi, Y., Okamura, T., Kohri, K., Kato, Y.: Detection and monitoring of simple renal cysts with computed tomography. Urol. Int. **54**(3), 128–131 (1995)
4. Ferjani, M.F., Ferjani, M.: Disease Prediction Using Machine Learning, vol. 10. Bournemouth University, Bournemouth (2020)
5. Rana, S., Jain, S., Virmani, J.: Classification of focal kidney lesions using wavelet-based texture descriptors (2016)
6. Mardison, Yuhandri: Detection of kidney cysts of kidney ultrasound image using hybrid method: KNN, GLCM, and ANN backpropagation. In: 2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE), pp. 222–227. IEEE (2023)
7. Qin, J., Chen, L., Liu, Y., Liu, C., Feng, C., Chen, B.: A machine learning methodology for diagnosing chronic kidney disease. IEEE Access **8**, 20991–21002 (2019)

8. Kline, T.L., et al.: Automatic semantic segmentation of kidney cysts in MR images of patients affected by autosomal-dominant polycystic kidney disease. Abdom. Radiol. **46**, 1053–1061 (2021)
9. Chen, G.: Prediction of chronic kidney disease using adaptive hybridized deep convolutional neural network on the internet of medical things platform. IEEE Access **8**, 100497–100508 (2020)
10. N. I. of Health Clinical Center (2018). https://nihcc.app.box.com/v/DeepLesion
11. Sudharson, S., Kokil, P.: Abnormality classification in the kidney ultrasound images using singular value decomposition features. In: 2019 IEEE Conference on Information and Communication Technology, pp. 1–5. IEEE (2019)
12. Hossain, T., Sayed, F., Islam, S.: Adaptive local binary pattern: a novel feature descriptor for enhanced analysis of kidney abnormalities in CT scan images using ensemble based machine learning approach. arXiv preprint arXiv:2404.14560 (2024)
13. Chowdhury, S.T., Mukhopadhyay, S., Narendra, K.S.: Mutual learning algorithm for kidney cyst, kidney tumor, and kidney stone diagnosis. In: 2023 18th Conference on Computer Science and Intelligence Systems (FedCSIS), pp. 401–410. IEEE (2023)
14. Hasan, M., Islam, M.N.: CT kidney dataset: normal-cyst-tumor and stone (2022)

Author Index

A
Adluri, Vijaya Lakshmi 90
Agrawal, Satyam 3
Ahmed, Rustom Ali 71
Akkasaligar, Prema T. 405
Anna Devi, E. 99
Antonijevic, Milos 350
Arega, Kedir Lemma 39
Arumugam, Sajeev Ram 99

B
Bacanin, Nebojsa 217, 350
Barad, Falguni Mukeshbhai 339
Beyene, Asrat Mulatu 39
Bhatt, Dhairya A. 379
Bhatt, Nikita 173
Bhatt, Nirav 173
Bhesaniya, Bhavin 305
Bune, Nurhussen Menza 39

C
Champaneria, Malav 173
Chintalapati, Kaushik 90
Chippalkatti, Srinivas 3
Chouhan, Yashoda Alpesh 397
Chudasama, Chetankumar 397

D
Darisi, Maadesh 90
Datta, Ashis 71
Dave, Harsh K. 17
Deshpande, Arati V. 3
Devapooja, John Paul 90
Dholakia, Neel H. 160
Diwan, Anjali 379
Dondapati, Mohan Aditya 90
Dubljanin, Dobrivoje 217
Duraisamy, S. 144

E
Elavarasi, J. 99

G
Ghosal, Palash 71

I
Ijeri, Dakshayani 364
Ingle, Anup 3

J
Jeyasudha, M. 232
Jhanwar, Kavita Yagyanarayan 326
Joshi, Akshit A. 17
Joshi, Devkumar R. 17
Jovanovic, Luka 217, 350

K
Kalariya, Shyam 111
Kansariwala, Hetashriben D. 17
Karuppasamy, Sankar Ganesh 99
Kothari, Nishant 160
Kozakijevic, Stanislava 350
Kulkarni, Pradnya V. 277

L
Ladva, Vipul 160
Litoriya, Ratnesh 59

M
Mahadeva, Rajesh 379
Makwana, Hemant 266
Mangal, Dharmendra 266
Meenakshi, A. 120
Mehta, Apurva. A. 294
Mirajkar, Sanjana 364
Mravik, Milos 350
Muneeswari, G. 256

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
S. Rajagopal et al. (Eds.): ASCIS 2024, CCIS 2426, pp. 417–418, 2025.
https://doi.org/10.1007/978-3-031-86296-0

N
Nimbai, Rahul 405
Njegus, Angelina 217

P
Panchal, Dhyani 173
Pandya, Swayam R. 17
Pate, Sanjay E. 187
Patel, Amit Khjimjibhai 339
Patel, Hardik Mahendrabhai 111
Patil, Ashish K. 133
Patil, Mayur 405
Patil, Pushpa B. 364
Pattar, Santosh 405
Perisic, Jasmina 350
Prajapati, Purvi 173
Prakash, G. 232
Prakash, S. 232
Purkayastha, Kunal 71
Purohit, Bravim K. 17

R
Radomirovic, Branislav 217
Rajpara, Harshil 379
Ramteke, Rakesh J. 187
Ranpariya, Akshay 160
Rathod, Veereshkumar 405
Revar, Ashish 305

S
Sadaria, Priti 326
Saravanan, P. 26
Shankar, R. 144
Sheela Gowr, P. 99
Shinde, Nikhil Mohan 3
Shukla, Madhu 160
Sinha, Rimjhim 277
Sipai, Uvesh 160
Sivaranjani, R. 144
Sivasakthi, M. 120
Spalevic, Petar 217
Srinivasa Rao, Gorapalli 256
Sutaria, Kamal 17
Syed, Simrin 160

T
Tariku, Wegderes 39
Tune, Kula Kekeba 39

V
Vadivazhagan, K. 26
Vanarase, Sagar Sudhakar 3
Verma, Deepak Kumar 397
Vyas, Malay 294

W
Wandra, Kalpesh 305

Y
Yadav, Deepak Kumar 133
Yele, Sachin 59

Z
Zivkovic, Miodrag 217, 350